AD-A089 169 HOUSTON UNIV TX DEPT OF ELECTRICAL ENGINEERING F/G 16/4
ANALYTICAL DESIGN OF TERMINALLY GUIDED MISSILES.(U)
JAN 80 L SHIEH DAAK40-79-C-0061
UNCLASSIFIED NL

1 OF 2
AD
A089169

Report DAAK 40-79-C-0061

LEVEL II

ANALYTICAL DESIGN OF TERMINALLY GUIDED MISSILES

AD A089169

Leang-San Shieh
Department of Electrical Engineering
University of Houston
Houston, Texas 77004

January 2, 1980

Final Report for period 1 February 1979 - 31 December 1979

Prepared for

U. S. ARMY MISSILE RESEARCH AND DEVELOPMENT COMMAND

Redstone Arsenal, Ala. 35809

80 7 3 013

15 Report DAAK 40-79-C-0061

6 ANALYTICAL DESIGN OF TERMINALLY GUIDED MISSILES.

10 Leang-San Shieh
Department of Electrical Engineering
University of Houston
Houston, Texas 77004

11 2 Jan[redacted]80

12 139

9 Final Report, [redacted] 1 Feb [redacted] - 31 Dec [redacted] 79,

Prepared for

U. S. ARMY MISSILE RESEARCH AND DEVELOPMENT COMMAND

Redstone Arsenal, Ala. 35809

401229

Report Documentation Page

1. Report Number:

   DAAK 40-79-C-0061

2. Govt. Acession No.

3. Recipient's Catalog Number

4. Title:

   Analytical Design of Terminally Guided Missiles

5. Type of Report and Period Covered

   Final Report for period 1 February 1979 - 31 December 1979

6. Performing Org. Report Number

7. Author:

   Leang-San Shieh

8. Contract or Grant Number

   DAAK 40-79-C-0061

9. Performing Organization Name and Address:

   Department of Electrical Engineering
   University of Houston
   Houston, Texas 77004

10. Program Element, Project, Task Area and Work Unit Numbers

11. Controlling Office Name and Address:

    Commander, US Army Missile R&D Command
    Attn: DRDMI-TGT
    Redstone Arsenal, Alabama 35809

12. Report Date:

January 2, 1980

13. Number of Pages

14. Monitoring Agency Name and Address

The U. S. Army Missile R&D Command,
Attn: DRDMI-TGT, Redstone Arsenal, Al. 35809

15. Security Class:

Unclassified

16. Distribution Statement (of this report)

17. Distribution Statement (of this abstract)

18. Supplementary Notes

19. Keywords:

Digital Control, Terminal Homing, Multivariable System, Design, Synthesis

20. Abstract

The newly developed dominant-data matching method, which was the research result supported by the U. S. Army Missile Research and Development Command under DAAK 40-78-C-0017, has been successfully extended to design a digital pitch control system of a semiactive terminal homing missile system. As a result, the designed digital controller can be implemented on a microprocessor. Also, a direct-decoupling method for multivariable control system designs has been developed and modified. This newly developed simple and practical method can be readily applied by a practicing control engineer for coupled high-order multivariable control system designs. The feasibility of synthesizing a multi-port controller without using integrators has been studied. As a result, a new matrix Sturm series and block canonical form of a matrix transfer function has been developed. The practical applications of the newly developed results on multi-port network synthesis are further investigated. Other new findings of this research are reported in the appendix.

ABSTRACT

The newly developed dominant-data matching method, which was the research result supported by the U. S. Army Missile Research and Development Command under DAAK 40-78-C-0017, has been successfully extended to design a digital pitch control system of a semiactive terminal homing missile system. As a result, the designed digital controller can be implemented on a microprocessor. Also, a direct-decoupling method for multivariable control system designs has been developed and modified. This newly developed simple and practical method can be readily applied by a practicing control engineer for coupled high-order multivariable control system designs. The feasibility of synthesizing a multi-port controller without using integrators has been studied. As a result, a new matrix Sturm series and block canonical form of a matrix transfer function has been developed. The practical applications of the newly developed results on multi-port network synthesis are further investigated. Other new findings of this research are reported in the appendix.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

This report deals with the digital redesign of the pitch control system of an unstable semi-active terminal homing missile system. It also concerns with the design of a coupled high-order multivariable control system and the realization of a multi-port controller.

In Chapter II we extend the newly developed dominant-data matching method, which was the research result supported by the U. S. Army Missile R&D Command under DAAK 40-78-C-0017, to design the digital controller for the pitch control system of an unstable semi-active terminal homing missile system. In Chapter III, we introduce a new and simple method for coupled high-order multivariable control system designs. In Chapter IV, we develop a new block canonical form of a matrix transfer function for possible realization of a multi-port controller without using integrators.

Other new findings of this research are reported in the appendix.

Chapter II

A DOMINANT-DATA MATCHING METHOD FOR DIGITAL FILTERS AND DIGITAL CONTROL SYSTEMS MODELING AND DESIGN

L. S. Shieh[1], Y. F. Chang[1], and R. E. Yates[2]

ABSTRACT

A dominant-data matching method is presented for obtaining a reduced-order discrete-data pulse-transfer function from either a high-order continuous-data transfer function or a high-order discrete-data pulse-transfer function, and for identifying the pulse-transfer function of a system from available experimental time and frequency response data. The method may also be applied to the digital control systems design problem with various sampling periods. The same method can be used for digital filter designs if the filter specifications obtained by this method are viewed as control specifications. The discrete-data system has the exact dominant characteristic performance of the original continuous-data or digital system. The relaxation of the sampling period requirement and the flexibility of our new method facilitate the practical industrial implementation and application.

[1]L. S. Shieh and Y. F. Chang are with the Department of Electrical Engineering, University of Houston, Houston, Texas 77004.

[2]R. E. Yates is with the Guidance and Control Directorate, U.S. Army Missile Command, Redstone Arsenal, Alabama 35809.

I. INTRODUCTION

Most practical industrial circuits and control systems are continuous-time systems for which analog filters and controllers are employed to improve performance. The recent availability of high performance, low cost microprocessors and associated electronics has led to replacement of many continuous systems with systems employing digital filters and controllers. Many techniques have been developed for digital control systems design [1]-[4]. Among them the ω-domain bilinear transformation is often applied to design industrial digital controllers. However, this method is graphical and involves "cut-and-try" procedures. Recently, Kuo [5] and others developed an optimal discrete-time data matching method for the redesign of a continuous-data system. Constant controllers instead of dynamic digital controllers are mainly employed in these designs. As a result, good performances of redesigned systems can be achieved if the frequency of the input signal is sufficiently lower than the sampling frequency. As an alternate to Kuo's time-domain approach, Rattan and Yeh [6] have given an elegant frequency-domain method for the redesign of continuous-data systems. The method of weighted least-squares complex-curve fitting due to Levy [7] and Sanathanan and Koerner [8] has been successfully extended in the z-domain to determine a dynamic digital controller. As a result of these efforts, better performance of redesigned systems can be achieved. However, this method is restricted to systems whose controllers are selected in such a way that the linear solution of the unknown constants in the controller is possible. On the other hand, if both feed-forward and feedback dynamic digital controllers are employed in the design, the closed-loop pulse-transfer function may have nonlinear coefficients of the unknown constants of the controllers. Therefore, the linear solution to their method may not be valid. Furthermore, most often design goals are assigned by using a mixture of time-domain, frequency-domain and complex-domain control specifications [9] rather than a set of frequency-response requirements. Therefore, complex-curve fitting

methods may not be applicable. In this paper, a computer-aided method is proposed for matching the dominant data of a high-order continuous-data system, or a discrete-data system, with the dominant response of a low-order digital system replacement. Also, methods are given for system identification and digital controller design of these systems.

II. DOMINANT DATA AND DOMINANT-DATA MATCHING METHOD

The characteristics of a control system or a filter are often expressed by either a time-response curve, or a frequency-response curve or a set of poles and zeros in the complex plane, or both. The quantitative description of the steady-state behavior is characterized by its final value as $t \to \infty$ and by the value of the steady-state frequency response as $\omega \to 0$. On the other hand, the quantitative description of transient behavior is represented by its time-domain control specification [9] (for example, the percentage overshoot and the rise time) and by the frequency-domain control specifications (for example, the maximum value of the closed-loop frequency response and the bandwidth). These specifications which are defined for control systems can be considered specifications of analog or digital filters. This is because a digital system (or a discrete-data system) can be viewed as a continuous-data system in the frequency domain when $z = e^{j\omega T}$ where T is a sampling period.

Some empirical observations or rules of thumb due to Axelby [10] that link the specifications of the continuous-data systems in both the time and frequency domains are as follows:

1. $$M_t \cong M_p \cong \frac{1}{\sin \phi_m} \qquad (1a)$$

M_t: Maximum value of unit-step response

M_p: Maximum value of the closed-loop frequency response

ϕ_m: Phase margin.

2. $M_e \cong \frac{1}{\omega_c}$ (1b)

M_e: Maximum value of the error of the unit-ramp function.

ω_c: Gain-crossover frequency

3. $\omega_p \cong \omega_c$ (1c)

ω_p: The peak value frequency or the frequency when M_p occurs.

4. $\dot{M}_t \cong \omega_c$ (1d)

$\dot{M}_t$: Maximum value of the unit-impulse response.

5. $t_p \cong \frac{3}{\omega_c}$ (1e)

t_p: The peak value time or the time when M_t occurs.

6. $t_v \cong \frac{1.8}{\omega_c}$ (1f)

t_v: The time when the maximum error of the ramp function with respect to its input occurs.

7. $t_c \cong \frac{1}{\omega_c}$ (1g)

t_c: The time when $\dot{M}_t$ occurs.

Other rules of thumb according to Truxal [11] are:

8. $t_r\omega_b \cong 0.6\pi$ to 0.9π (1h)

t_r: The rise time or the time required for the response to go from 10 to 90 percent of its final value.

ω_b: The bandwidth in rad/sec.

9. $t_d \cong \frac{1}{K_v}$ (1i)

t_d: The delay time or the time required to reach 50 percent of its final value.

K_v: The velocity error constant.

The above rules can be verified by using the standard second order transfer function:

$$\frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{2}$$

where R(s) and Y(s) are the input and output functions, respectively, ξ is the damping ratio, and ω_n is the undamped natural angular frequency. The zero of the system is located at infinity, and the finite poles are in the complex plane. The ξ and ω_n as used herein are defined as the complex-domain control specifications.

Recently Shieh et al. [12] have studied the relationships between the complex-domain specifications (ξ and ω_n) and the time-domain and frequency-domain specifications by using a more sophisticated model

$$\frac{Y(s)}{R(s)} = \frac{\tau\omega_n s + \omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} = \frac{\tau\left(\frac{s}{\omega_n}\right) + 1}{\left(\frac{s}{\omega_n}\right)^2 + 2\xi\left(\frac{s}{\omega_n}\right) + 1} = \frac{\tau s^* + 1}{s^{*2} + 2\xi s^* + 1} \tag{3}$$

where $s^* = s/\omega_n$ is a normalized complex variable and τ indicates the location of a finite zero. The relationships are as follows:

1. M_t, t_p, ξ, ω_n, and τ

$$t_p = \left(\pi + \tan^{-1}\frac{\tau\sqrt{1-\xi^2}}{\tau\xi - 1}\right)\Bigg/\left(\omega_n\sqrt{1-\xi^2}\right) \tag{4a}$$

$$M_t = 1 + e^{-\xi\omega_n t_p}\left(\tau^2 - 2\tau\xi + 1\right)^{1/2} \tag{4b}$$

2. M_p, ω_p, ξ, ω_n, and τ

$$\omega_p = \omega_n\sqrt{1 - 2\xi^2} \qquad \text{if } \tau = 0 \tag{4c}$$

$$M_p = 1/(2\xi\sqrt{1-\xi^2}) \qquad \text{if } \tau = 0 \tag{4d}$$

and

$$\omega_p = \frac{\omega_n}{\tau}\left[-1 + \sqrt{(\tau^2 + 1)^2 - 4\tau^2\xi^2}\right]^{1/2} \quad \text{if } \tau \neq 0 \tag{4e}$$

$$M_p = \frac{\tau^2}{\sqrt{2}}\left[\sqrt{\left(\tau^2 + 1\right)^2 - 4\xi^2\tau^2} - \left(\tau^2 + 1\right) + 2\xi^2\tau^2\right]^{-1/2} \quad \text{if } \tau \neq 0 \tag{4f}$$

3. ϕ_m, ω_c, ξ, ω_n, and τ

$$\phi_m = \tan^{-1}\left[\frac{\left(\frac{\omega_c}{\omega_n}\right)\tau + \left(2\xi - \tau\right)\left(\frac{\omega_n}{\omega_c}\right)}{1 - (2\xi - \tau)\tau}\right] \tag{4g}$$

$$\omega_c = \omega_n\left[2\xi\tau - 2\xi^2 + \sqrt{(2\xi^2 - 2\xi\tau)^2 + 1}\right]^{1/2} \tag{4h}$$

4. t_v, M_e, ξ, ω_n, and τ

$$t_v = \frac{1}{\omega_n\sqrt{1 - \xi^2}} \tan^{-1}\left[\frac{\sqrt{1 - \xi^2}}{\tau - \xi}\right] \tag{4i}$$

$$M_e = \left[2\xi - \tau + \sqrt{(1 + \tau^2 - 2\tau\xi)}\; e^{-\xi\omega_n t_v}\right]\Big/\omega_n \tag{4j}$$

5. t_c, $\dot{M}_t$, ξ, ω_n, and τ

$$t_c = \frac{1}{\omega_n\sqrt{1 - \xi^2}} \tan^{-1}\left[\frac{(1 - 2\xi\tau)\sqrt{1 - \xi^2}}{\xi - 2\tau\xi^2 + \tau}\right] \tag{4k}$$

$$\dot{M}_t = \omega_n e^{-\xi\omega_n t_c}\sqrt{\tau^2 - 2\xi\tau + 1} \tag{4l}$$

6. K_v, ξ, ω_n, and τ

$$K_v = \frac{\omega_n}{2\xi - \tau} \quad \text{if } \tau < 2\xi \tag{4m}$$

7. ω_b, ξ, ω_n, and τ

$$\omega_b = \omega_n\left[\left(1 + \tau^2 - 2\xi^2\right) + \sqrt{\left(1 + \tau^2 - 2\xi^2\right)^2 + 1}\right]^{1/2} \tag{4n}$$

The above analytical expressions can be plotted by letting $\omega_n = 1$ and ξ the variable. Once a specification is defined, other corresponding specifications can then be determined. In other words, if time-domain specifications are assigned, the corresponding frequency-domain and complex-domain specifications can be determined. The frequency-response data or the equivalent data obtained from Eqs. (1) and (4) at ω_n, ω_c, ω_π, ω_p, ω_b, etc., are the dominant data as used in this paper. Other important data will be the frequency-response at $s=j\omega=j0$ because these characterize steady-state behavior.

Our new dominant-data matching method matches the above dominant data of a continuous-data or a discrete-data system to those of the newly designed or modeled discrete-data system. The steps involved are as follows:

<u>Step 1:</u> Determine a set of dominant frequency-response data from the assigned time-domain and complex-domain specifications by using the rules and results in Eqs. (1) and (4).

<u>Step 2:</u> Assume a fixed configuration digital system and controllers with unknown constants. Determine the open-loop and the overall pulse-transfer function of the system.

<u>Step 3:</u> Formulate a set of linear/nonlinear equations by matching the unknown constants of the pulse-transfer function and the assigned dominant data. Solve the equations by using the multidimensional Newton-Raphson method [13], available as a library computer program package (called the Z system) in many digital computers [14].

Step 4: Estimate initial value for the numerical solution of the Newton-Raphson method by constructing a crude pulse-transfer function which can be obtained by a complex-curve fitting method.

Step 5: Compare the results with the assigned specifications.

When the dominant data are obtained from either a high-order continuous-data transfer function or a high-order discrete-data pulse-transfer function and a low-order pulse-transfer function is required; this is a model reduction problem. If the dominant data are determined from an experimental set of time and frequency data and the corresponding pulse-transfer function is required, this is the identification problem. The order of the identified pulse-transfer function depends on the number of dominant data parameters used. Therefore, the identified pulse-transfer function could be the reduced-order model of the original high-order system. The above two problems can be considered the modeling problem. When the design goals are specified by a set of dominant data and the digital controllers with unknown constants are designed to match the desired dominant data, this is the design problem for digital control systems. Applications of the above new method will be described in the following sections.

III. MODELING A REDUCED-ORDER PULSE-TRANSFER FUNCTION

We use a real stabilized pitch control system of a semiactive terminal homing missile [15] as an illustrative model to show that the characteristics of the transient-state response of a system can be estimated from the dominant frequency-response data and the applications of the proposed method to the identification and model reduction problems. A block diagram of the missile system is shown in Fig. 1. The closed-loop high-order transfer function is

$$\frac{Y(s)}{R(s)} = \frac{G_c(s)G_o(s)}{1 + G_c(s)G_o(s)H_g(s)} = \frac{G_e(s)}{1 + G_e(s)} \triangleq T_e(s) \tag{5a}$$

where

$G_c(s)$ = the stabilization filter

$$= \frac{1.6\left(\frac{s}{25} + 1\right)\left(\frac{s}{125} + 1\right)}{\left[\left(\frac{s}{150}\right)^2 + \left(\frac{0.6}{150}\right)s + 1\right]\left[\left(\frac{s}{200}\right)^2 + \left(\frac{0.8}{200}\right)s + 1\right]} \qquad (5b)$$

$G_o(s)$ = The transfer function of the actuator and aerodynamics of the missile system

$$= \frac{324332.316(s + 0.1933)(s + 65)(s + 1500)}{s(s - 2.921)(s + 3.175)(s + 87.9 \pm j95.5)(s + 112.5)(s + 138} \qquad (5c)$$

$H_g(s)$ = The transfer function of the gyro = 1 (5d)

$G_e(s) \triangleq G_c(s)G_o(s)$ = The unstable open-loop transfer function of the existing stabilized system. (5e)

The closed-loop transfer function $T_e(s)$ becomes

$$T_e(s) = \frac{b_0 s^{10} + b_1 s^9 + \ldots + b_9 s + b_{10}}{a_0 s^{11} + a_1 s^{10} + \ldots + a_{10} s + a_{11}} \qquad (6)$$

where

$a_0 = 1$ | $b_0 = 0$

$a_1 = 1.923554000 \times 10^3$ | $b_1 = 0$

$a_2 = 9.316239040 \times 10^5$ | $b_2 = 0$

$a_3 = 2.976950696 \times 10^8$ | $b_3 = 0$

$a_4 = 6.231675318 \times 10^{10}$ | $b_4 = 0$

$a_5 = 9.360329977 \times 10^{12}$ | $b_5 = 1.494523312 \times 10^{11}$

$a_6 = 9.749923212 \times 10^{14}$ | $b_6 = 2.563396371 \times 10^{14}$

$a_7 = 6.667397031 \times 10^{16}$ | $b_7 = 5.017212044 \times 10^{16}$

$a_8 = 2.420405431 \times 10^{18}$ | $b_8 = 2.926344345 \times 10^{18}$

$a_9 = 2.911920560 \times 10^{18}$ | $b_9 = 4.610004670 \times 10^{19}$

$a_{10} = 2.419047424 \times 10^{19}$ | $b_{10} = 8.802158509 \times 10^{18}$

$a_{11} = 8.802158509 \times 10^{18}$

The Nyquist plots of $G_e(s)$ and $G_o(s)$ are shown in Fig. 2. The dominant data of $G_e(s)$ are:

1. Real and imaginary parts of $G_e(s)$ at $s = j\omega = j0$ are

$$\mathrm{Re}\left[G_e(j0)\right] = -2.103817 \tag{7a}$$

$$\mathrm{Im}\left[G_e(j0)\right] = \infty$$

2. Gain margin G_{em} of this system $G_e(j\omega_\pi)$ is

$$G_{em} = \left|\frac{1}{G_e(j\omega_\pi)}\right| = \left|\frac{1}{\mathrm{Re}\,[G_e(j\omega_\pi)]}\right| \cong \left|\frac{1}{-1.5}\right| \tag{7b}$$

where the phase-crossover frequency ω_π is given by

$$\omega_\pi \cong 1.9 \text{ rad/sec, such that } \angle G_e(j\omega_\pi) = -180° \tag{7c}$$

The equivalent real and imaginary parts of $G_e(j\omega_\pi)$ at $\omega_\pi = 1.9$ rad/sec are

$$\mathrm{Re}\left[G_e(j\omega_\pi)\right] = -1.507944 \tag{7d}$$

$$\mathrm{Im}\left[G_e(j\omega_\pi)\right] = -0.006490205 \tag{7e}$$

3. Phase margin ϕ_{em} of the system $G_e(j\omega)$ is

$$\phi_{em} = 180° + \angle G_e(j\omega_c) \cong 5.7787° \tag{7f}$$

where the gain crossover frequency ω_c is given by $\omega_c \cong 3.2$ rad/sec so that,

$$|G_e(j\omega_c)| = 1 \tag{7g}$$

Equivalent real and imaginary parts of $G_e(j\omega_c)$ are

$$\mathrm{Re}\left[G_e(j\omega_c)\right] = -0.9939143 \tag{7h}$$

$$\mathrm{Im}\left[G_e(j\omega_c)\right] = -0.09547478 \tag{7i}$$

It is required to determine a reduced-order pulse-transfer function such that the characteristics of the identified discrete-data model agree as closely as possible with those of the high-order continuous-data system.

Let the required overall pulse-transfer function be

$$T_r(z) = \frac{G_r(z)}{1 + G_r(z)} \tag{8a}$$

where the open-loop pulse-transfer function $G_r(z)$ is

$$G_r(z) = \frac{x_0 z^p + x_1 z^{p-1} + \ldots + x_{p-1} z + x_p}{(z - 1)(y_0 z^q + y_1 z^{q-1} + \ldots + y_{q-1} z + y_q)} \tag{8b}$$

$G_r(z)$ is assigned to be a type "1" system because $G_e(s)$ in Eq. (5e) is a type "1" system. To match the five dominant data in Eq. (7) we choose $q = 2$, $p = 2$, and $y_0 = 1$. Thus $G_r(z)$ becomes

$$G_r(z) = \frac{x_0 z^2 + x_1 z + x_2}{(z - 1)(z^2 + y_1 z + y_2)} \tag{9}$$

where x_ℓ and y_ℓ in Eq. (9) are unknown constants to be determined. The goal is to determine the unknown constants x_ℓ and y_ℓ in $G_r(z)$ so that $G_r(z)$ as $z = e^{j\omega T}$ matches the dominant data in Eq. (7). The sampling period T (=0.008 sec) and ω_s(=250π rad/sec) are chosen to be synchronized with the 125 Hz pulse-width modulated actuator [16]. Since $G_e(s)$ and $G_r(z)$ are type "1" systems and we need to match the dominant data of $G_e(j\omega)$ as $\omega = 0$ in Eq. (7a) and the dominant data of $G_r(z)$ as $\omega = 0$ or $z = e^{j\omega T} = 1$ in Eqs. (8b) or (9), the expression for $G_r(z)$ in Eq. (8b) is modified as follows:

Substituting $z = z^* + 1$ into Eq. (8b) yields

$$G_r(z^*) = \frac{x_p^* + x_{p-1}^* z^* + \ldots + x_0^* z^{*p}}{z^*(y_q^* + y_{q-1}^* z^* + \ldots + y_0^* z^{*q})}$$

$$= e_{-1} z^{*-1} + e_0 + e_1 z^* + e_2 z^{*2} + \ldots \tag{10a}$$

where

$$x_p^* = \sum_{i=0}^{p} x_i,\ x_{p-1}^* = \sum_{i=1}^{p} i x_{p-i},\ y_q^* = \sum_{i=0}^{q} y_i,\ y_{q-1}^* = \sum_{i=1}^{q} i y_{q-i},$$

$$e_{-1} = x_p^*/y_q^* \text{ and } e_0 = (y_q^* x_{p-1}^* - y_{q-1}^* x_p^*)/y_q^{*2}, \text{ etc.}$$

Equating the respective real and imaginary parts of $G_r(z)$ for $\omega = 0$ and those of $G_r(z^*)$ for $\omega = 0$ gives

$$\mathrm{Re}\left[G_r(z)\right]\Big|_{z = 1 + j0} = \mathrm{Re}\left[G_r(z^*)\right]\Big|_{z^* = z-1 = j0} = e_0 \tag{10b}$$

and

$$\mathrm{Im}\left[G_r(z)\right]\Big|_{z = 1 + j0} = \mathrm{Im}\left[G_r(z^*)\right]\Big|_{z^* = z-1 = j0} = \infty \tag{10c}$$

Eqs. (10b) and (10c) imply that e_0 in Eq. (10b) is the asymptotic line of the type "1" systems at low frequencies.

In the frequency domain, Eq. (9) can be expressed in an alternative form as follows:

Let us define

$$z = e^{j\omega_k T} = \cos \omega_k T + j \sin \omega_k T \triangleq u_k + jv_k \tag{11a}$$

and substituting $z = u_k + jv_k$ into Eq. (9), we have

$$G_r(u_k,v_k) = \frac{(x_0 u_k^2 - x_0 v_k^2 + x_1 u_k + x_2) + j(2x_0 u_k v_k + x_1 v_k)}{\left[(u_k - 1)(u_k^2 - v_k + y_1 u_k + y_2) - v_k(2u_k v_k + y_1 v_k)\right] + j\left[(u_k - 1)(2u_k v_k + y_1 v_k) + v_k(u_k^2 - v_k^2 + y_1 u_k + y_2)\right]}$$

$$= R_k + j\, I_k \tag{11b}$$

where ω_k are specific frequencies and $R_k \triangleq \mathrm{Re}\left[G_r(u_k,v_k)\right]$, $I_k \triangleq \mathrm{Im}\left[G_r(u_k,v_k)\right]$. If R_k and I_k are the known or assigned values at frequencies ω_k, we can obtain two linear equations. First, we multiply both sides of Eq. (11b) by the common denominator, then we separate the real and imaginary parts and then equate the respective real and imaginary parts. Thus we have

$$\begin{aligned} f_i(x_0,x_1,x_2,y_1,y_2) = {} & (x_0 u_k^2 - x_0 v_k^2 + x_1 u_k + x_2) \\ & - R_k\left[(u_k - 1)(u_k^2 - v_k^2 + y_1 u_k + y_2)\right. \\ & \left. - v_k(2u_k v_k + y_1 v_k)\right] + I_k\left[(u_k - 1)(2u_k v_k\right. \\ & \left. + y_1 v_k) + v_k(u_k^2 - v_k^2 + y_1 u_k + y_2)\right] = 0 \end{aligned} \tag{11c}$$

and

$$f_{i+1}(x_0,x_1,x_2,y_1,y_2) = (2x_0u_kv_k + x_1v_k) - R_k\left[(u_k - 1)(2u_kv_k + y_1v_k) + v_k(u_k{}^2 - v_k{}^2 + y_1u_k + y_2)\right] - I_k\left[(u_k - 1)(u_k{}^2 - v_k{}^2 + y_1u_k + y_2) - v_k(2u_kv_k + y_1v_k)\right] = 0 \tag{11d}$$

Using the expressions in Eqs. (10b), (11c), and (11d) and the assigned dominant data in Eq. (7) we can formulate one nonlinear equation and four linear equations $f_i(x_\ell,y_\ell) = 0$ for $i = 1,2,\ldots,5$ as follows:

(i) The data in Eq. (7a), or $\mathrm{Re}\left[G_e(j\omega)\right] = -2.103$ for $\omega = 0$ and the relationship in Eq. (10b) gives a nonlinear equation:

$$f_1(x_\ell,y_\ell) = (2x_0 + x_1)(1 + y_1 + y_2) - (2 + y_1)(x_0 + x_1 + x_2) + 2.103(1 + y_1 + y_2)^2 = 0 \tag{12}$$

(ii) From Eqs. (7d) and (7e), $\mathrm{Re}\left[G_e(j\omega_\pi)\right] = -1.507944$ and $\mathrm{Im}\left[G_e(j\omega_\pi)\right] = -0.00649025$ for $\omega_\pi = 1.9$ rad/sec. We define

$$R_k \triangleq R_{1.9} = -1.507944,\ I_k \triangleq I_{1.9} = -0.006490205,\ \omega_k \triangleq \omega_{1.9} = \omega_\pi = 1.$$

$$u_k \triangleq u_{1.9} = \cos\omega_{1.9}T = 0.99988448 \text{ and } v_k \triangleq v_{1.9} = \sin\omega_{1.9}T = 0.01519941 \text{ as } T = 0.008 \text{ sec}$$

Substituting the above data into Eqs. (11c) and (11d) and letting $i = 2$ we have two linear equations $f_2(x_\ell,y_\ell) = 0$ and $f_3(x_\ell,y_\ell) = 0$ as shown in Eqs. (11c) and (11d).

(iii) From Eqs. (7h) and (7i) we define $\omega_k = \omega_c = 3.2 \triangleq \omega_{3,2}$, $u_k \triangleq u_{3.2} = \cos\omega_{3.2}T = 0.99967234$, $v_k \triangleq v_{3.2} = \sin\omega_{3.2}T = 0.025597204$, $R_k \triangleq R_{3.2} = -0.9939143$ and $I_k \triangleq I_{3.2} = -0.09547478$. Substituting the above data and $i = 4$ into Eqs. (11c) and (11d) yields two more linear equations $f_4(x_\ell,y_\ell) = 0$ and $f_5(x_\ell,y_\ell) = 0$. Thus, we have five simultaneous equations $f_i(x_\ell,y_\ell) = 0$ with five unknown

constants x_ℓ and y_ℓ to be solved. Notice that if the data of Eqs. (7b), (7c), (7f), and (7g) are used to match the unknown coefficients of Eq. (11b), the resulting equations $f_i(x_\ell, y_\ell) = 0$ in general are nonlinear. Therefore, we note that in general the equations $f_i(x_\ell, y_\ell) = 0$ are nonlinear. The Newton-Raphson method available as a library computer program in most digital computers [14] can be applied to solve these nonlinear equations. However, as is well known, the Newton-Raphson method will converge to a desired solution for a small range of starting values or initial solution estimates. To improve the convergence and to obtain the set of desired solutions, we offer the following method for initial estimates.

Since $f_1(x_\ell, y_\ell) = 0$ is nonlinear and $f_i(x_\ell, y_\ell) = 0$, $i = 2,\ldots,5$ are linear equations, we linearize $f_1(x_\ell, y_\ell) = 0$ by choosing a very low frequency. For example, if we choose $\omega_k = 0.01 \triangleq \omega_{0.01}$, then $R_k \triangleq R_{0.01} = -2.1$, $I_k \triangleq I_{0.01} = 40.17319$, $u_k \triangleq u_{0.01} = \cos \omega_{0.01}T = 0.99999950$ and $v_k \triangleq v_{0.01} = \sin \omega_{0.01}T = 8 \times 10^{-5}$. Solving $f_1^*(x_\ell, y_\ell) = 0$ and $f_i(x_\ell, y_\ell) = 0$, $i = 2,\ldots,5$ for the unknown constants x_ℓ (defined as x_ℓ^*) and y_ℓ (defined as y_ℓ^*) we get $x_0^* = 0.00679254$, $x_1^* = -0.0123359$, $x_2^* = 0.00554537$, $y_1^* = -1.9985417$, and $y_2^* = 0.99794573$. Using these values as initial estimates for the solutions of $f_i(x_\ell, y_\ell) = 0$ using the Newton-Raphson method we obtain the solution $x_0 = 0.00679259$, $x_1 = -0.01233599$, $x_2 = 0.00554531$, $y_1 = -1.9985412$, and $y_2 = 0.9979452$ at the second iteration with error tolerance of 10^{-6}. The desired open-loop pulse-transfer function is

$$G_r(z) = \frac{0.006792596z^2 - 0.012335992z + 0.0055453114}{z^3 - 2.9985412z^2 + 2.9964864z - 0.9979452} \tag{13}$$

A Nyquist plot of $G_r(z)$ is shown in Fig. 2. The plot matches closely that of $G_e(s)$ not only at the dominant frequencies but also at others. The $G_r(z)$ is seen to be a good reduced model of the original unstable system $G_e(s)$. This is the contribution

of our new method because there are no known effective model-reduction methods for unstable systems. The resulting closed-loop pulse-transfer function which is the *reduced-order discrete-data model* of the original high-order continuous data system is

$$T_r(z) = \frac{G_r(z)}{1 + G_r(z)} = \frac{0.006792596z^2 - 0.012335992z + 0.0055453114}{z^3 - 2.991748524z^2 + 2.984150408z - 0.9923998886} \tag{14}$$

Since the assigned dominant data are the steady-state frequency response, it is interesting to compare responses of $T_e(s)$ in Eq. (6) and $T_r(z)$ in Eq. (14) shown in Fig. 3. Observe that both the transient response and steady-state response of the reduced-order model $T_r(z)$ are excellent matches of the original high-order system. This indicates that the dynamic characteristics of the system (for example, peak value time and overshoot, which may not occur at the sampling time) are indirectly controlled by the assignment of the gain-crossover frequency and the phase margin. This is a major advantage of our new method. Also note that the reduced-order model gives an excellent approximation of the original system when driven by high-frequency input signals.

To determine the initial estimates x_ℓ^* and y_ℓ^*, a general formulation of a set of linear equations can be constructed from the following complex-curve fitting method.

Consider the pulse-transfer function

$$G(z) = \frac{x_0^* z^m + x_1^* z^{m-1} + \dots + x_m^*}{y_0^* z^n + y_1^* z^{n-1} + \dots + y_n^*} \tag{15a}$$

where $y_0^* = 1$ and x_ℓ^* and y_ℓ^* are unknown constants to be determined. Substituting $z^r = e^{jr\omega_k T} = \cos r\omega_k T + j \sin r\omega_k T$ into Eq. (15a) gives

$$G\left(e^{jr\omega_k T}\right) = \frac{\sum_{\ell=0}^{m} x_\ell^{*} \cos(m - \ell)\omega_k T + j \sum_{\ell=0}^{m} x_\ell^{*} \sin(m - \ell)\omega_k T}{\sum_{\ell=0}^{n} y_\ell^{*} \cos(n - \ell)\omega_k T + j \sum_{\ell=0}^{n} y_\ell^{*} \sin(n - \ell)\omega_k T}$$

$$= R(\omega_k) + jI(\omega_k) = R_k + jI_k \tag{15b}$$

where R_k and I_k are the real and imaginary parts of the transfer function at the experimental frequencies or the assigned frequencies ω_k. After multiplying both sides of Eq. (15b) by the common denominator and separating the real and imaginary parts, we equate the respective real and imaginary parts. This yields the following matrix equation:

$$\begin{bmatrix} \cos m\omega_1 T & \cos(m-1)\omega_1 T & \cdots & 1 & (-R_1\cos(n-1)\omega_1 T + I_1\sin(n-1)\omega_1 T) & \cdots & (-R_1\cos\omega_1 T + I_1\sin\omega_1 T) & -R_1 \\ \sin m\omega_1 T & \sin(m-1)\omega_1 T & \cdots & 0 & (-R_1\sin(n-1)\omega_1 T - I_1\cos(n-1)\omega_1 T) & \cdots & (-R_1\sin\omega_1 T - I_1\cos\omega_1 T) & -I_1 \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \\ \cos m\omega_i T & \cos(m-1)\omega_i T & \cdots & 1 & (-R_i\cos(n-1)\omega_i T + I_i\sin(n-1)\omega_i T) & \cdots & (-R_i\cos\omega_i T + I_i\sin\omega_i T) & -R_i \\ \sin m\omega_i T & \sin(m-1)\omega_i T & \cdots & 0 & (-R_i\sin(n-1)\omega_i T - I_i\cos(n-1)\omega_i T) & \cdots & (-R_i\sin\omega_i T - I_i\cos\omega_i T) & -I_i \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \end{bmatrix} \begin{bmatrix} x_0^{*} \\ x_1^{*} \\ \vdots \\ x_m^{*} \\ y_1^{*} \\ y_2^{*} \\ \vdots \\ y_n^{*} \end{bmatrix}$$

$$= \begin{bmatrix} (R_0\cos n\omega_0 T - I_0\sin n\omega_0 T) \\ (R_0\sin n\omega_0 T + I_0\cos n\omega_0 T) \\ \vdots \\ (R_i\cos n\omega_i T - I_i\sin n\omega_i T) \\ (R_i\sin n\omega_i T + I_i\cos n\omega_i T) \\ \vdots \end{bmatrix} \tag{15c}$$

Substituting the selected $(n + m + 1)$ frequency-response data into Eq. (15c), we can solve for the required $(n + m + 1)$ unknown constants x_ℓ^{*} and y_ℓ^{*}.

IV. DIGITAL CONTROL SYSTEM DESIGN

Consider the pitch control transfer function of the missile system of Eq. (5a). The unity-feedback system without the stabilization filter $G_c(s)$ is unstable, and a rate gyro is not available for this example system. It is required to design a digital controller $G_c(z)$ instead of an analog controller $G_c(s)$ such that the designed system has the exact control specifications [9] of the original stabilized continuous-data system given in Eq. (7). Furthermore, the response $G_e(j\omega)$ at $\omega = 140$ rad/sec $\triangleq$ ω_{140} is chosen as a dominant data constraint because the system has an inherent high frequency noise component at ω_{140}. This is a digital redesign problem. The structure of the digital control system is shown in Fig. 4. The closed-loop pulse-transfer function of the desired digital system becomes

$$\frac{Y(z)}{R(z)} = \frac{G_c(z)G_nG_o(z)}{1 + G_c(z)G_nG_o(z)} \triangleq T_{e1}(z) = \frac{G_{e1}(z)}{1 + G_{e1}(z)} \tag{16}$$

where

$$G_nG_o(z) = (1 - z^{-1})\mathcal{Z}\left[\frac{1}{s} G_o(s)\right] = \frac{b_0z^6 + b_1z^5 + \ldots + b_6}{a_0z^7 + a_1z^6 + \ldots + a_7}$$

$b_0 = 0.4095517916 \times 10^{-4}$ $\quad$ $a_0 = 1.0$

$b_1 = 0.2526111734 \times 10^{-3}$ $\quad$ $a_1 = -4.120000127$

$b_2 = -0.2575534058 \times 10^{-3}$ $\quad$ $a_2 = 6.894911119$

$b_3 = -0.1540502340 \times 10^{-3}$ $\quad$ $a_3 = -6.064285805$

$b_4 = 0.1096039580 \times 10^{-3}$ $\quad$ $a_4 = 3.023078996$

$b_5 = 0.8808224830 \times 10^{-5}$ $\quad$ $a_5 = -0.83313251$

$b_6 = -0.1358278543 \times 10^{-9}$ $\quad$ $a_6 = 0.09942985948$

$a_7 = -0.1532773068 \times 10^{-5}$

$G_{e1}(z) \triangleq G_c(z)G_nG_o(z)$

$G_n(s) = \dfrac{1 - e^{-sT}}{s}$ = the zero-order hold

T = the sampling period = 0.008 sec.

$G_c(z)$ is the desired digital controller and is

$$G_c(z) = \frac{x_0 z^2 + x_1 z + x_2}{z^2 + y_1 z + y_2} \tag{17}$$

where x_ℓ and y_ℓ are unknown constants to be determined. Because $G_c(z)$ is a forward controller, the equations $f_i(x_\ell, y_\ell) = 0$ can be formulated from the following equations:

$$G_c(z)\Big|_{z = e^{j\omega_k T}} = \frac{G_e(s)\big|_{s = j\omega_k}}{G_n G_o(z)\big|_{z = e^{j\omega_k T}}} \tag{18}$$

where $\omega_0 = 0$, $\omega_{1.9} = 1.9$, $\omega_{3.2} = 3.2$, and $\omega_{140} = 140$. Notice that $G_n G_o(z)$ for $z = e^{j\omega_k T}$ is not equal to $G_o(s)$ for $s = j\omega_k$ unless $T = 0$. Using the dominant data of Eq. (7), the required response at ω_{140}, and the relationships expressed in Eqs. (10b) and (18) yields a set of equations $f_i(x_\ell, y_\ell) = 0$ for $i = 1,2,\ldots,5$ as follows:

(i) Using Eqs. (7a) and (10b) when $\omega = \omega_0 = 0$, yields a nonlinear equation;

$$\begin{aligned} f_1(x_\ell, y_\ell) = {} & -4.557577105 \times 10^{-8}(x_0 + x_1 + x_2)(1 + y_1 + y_2) \\ & - 7.016133905 \times 10^{-11}\Big[(2x_0 + x_1)(1 + y_1 + y_2) \\ & - (2 + y_1)(x_0 + x_1 + x_2)\Big] + 2.1 \times 3.505134712 \times 10^{-8} \\ & (1 + y_1 + y_2)^2 = 0 \end{aligned} \tag{19a}$$

(ii) Using Eqs. (7h), (7i), and (18) when $\omega = \omega_{3.2} = 3.2$ we get:

$$\mathrm{Re}\Big[G_c(e^{j\omega_{3.2}T})\Big] = 1.5987861 \text{ and } \mathrm{Im}\Big[G_c(e^{j\omega_{3.2}T})\Big] = 0.23560917$$

The resulting linear equation is

$$\begin{aligned} f_2(x_\ell, y_\ell) = {} & (x_0 u_{3.2}^2 - x_0 v_{3.2}^2 + x_1 u_{3.2} + x_2) \\ & - 1.5987861(u_{3.2}^2 - v_{3.2}^2 + y_1 u_{3.2} + y_2) \\ & + 0.23560917(2u_{3.2} v_{3.2} + y_1 v_{3.2}) = 0 \end{aligned} \tag{19b}$$

where $u_{3.2} = 0.99967234$ and $v_{3.2} = 0.025597204$

(iii) Using Eqs. (18) and (19b) for $\omega = \omega_{3.2} = 3.2$, obtain a linear equation

$$f_3(x_\ell, y_\ell) = (2x_0 u_{3.2} v_{3.2} + x_1 v_{3.2}) - 1.5987861(2u_{3.2} v_{3.2} + y_1 v_{3.2}) - 0.23560917(u_{3.2}^2 - v_{3.2}^2 + y_1 u_{3.2} + y_2) = 0 \quad (19c)$$

(iv) From Eq. (18) for $\omega = \omega_{140} = 140$ we have

$\mathrm{Re}\left[G_c(e^{j\omega_{140}T})\right] = 26.951878$ and $\mathrm{Im}\left[R_e G_c(e^{j\omega_{140}T})\right] = 19.196865$

The resulting linear equation is

$$f_4(x_\ell, y_\ell) = (x_0 u_{140}^2 - x_0 v_{140}^2 + x_1 u_{140} + x_2) - 26.951878(u_{140}^2 - v_{140}^2 + y_1 u_{140} + y_2) + 19.196865(2u_{140} v_{140} + y_1 v_{140}) = 0 \quad (19d)$$

where $u_{140} = 0.43568245$ and $v_{140} = 0.90010044$.

(v) From Eq. (19d) for $\omega = \omega_{140}$, we obtain another linear equation

$$f_5(x_\ell, y_\ell) = (2x_0 U_{140}^2 + x_1 v_{140}) - 26.951878(2u_{140} v_{140} + y_1 v_{140}) - 19.196865(u_{140}^2 - v_{140}^2 + y_1 u_{140} + y_2) = 0 \quad (19e)$$

The above set of linear and nonlinear equations can be solved using the Newton-Raphson method. The initial estimates for the Newton-Raphson solution may be determined from Eq. (15c). Another linear equation $f_1^*(x_\ell, y_\ell) = 0$, instead of $f_1(x_\ell, y_\ell) = 0$ in Eq. (19a), can be constructed to yield five linear equations with five unknown constants (x_ℓ^* and y_ℓ^*). $G_e(j\omega)$ at $\omega = 0.01 \triangleq \omega_{0.01}$ is used in this case. Substituting $\mathrm{Re}\left[G_c(e^{j\omega_{0.01}T})\right] = 1.5961120$ and $\mathrm{Im}\left[G_c(e^{j\omega_{0.01}T})\right] = 6.409642$ into Eq. (18) we get the linear equation

$$f_1^*(x_\ell, y_\ell) = (x_0 u_{0.01}^2 - x_0 v_{0.01}^2 + x_1 u_{0.01} + x_2)$$
$$- 1.5961120(u_{0.01}^2 - v_{0.01}^2 + y_1 u_{0.01} + y_2)$$
$$+ 6.409642(2u_{0.01}v_{0.01} + y_1 v_{0.01}) = 0 \qquad (20)$$

where $u_{0.01} = 0.99999950$ and $v_{0.01} = 8 \times 10^{-5}$.

Solving $f_i(x_\ell, y_\ell) = 0$, $i = 2,\ldots,5$ in Eq. (19) and $f_1^*(x_\ell, y_\ell) = 0$ in Eq. (20) gives a set of initial values x_ℓ^* and y_ℓ^*; $x_0^* = 2.9025918$, $x_1^* = 9.3580336$, $x_2^* = -10.066413$, $y_1^* = -0.42702018$, and $y_2^* = 0.80224909$. Using these initial values and the Newton-Raphson method to solve the equations in Eq. (19) we obtain the solution: $x_0 = 11.869083$, $x_1 = -13.49237$, $x_2 = 3.0584008$, $y_1 = -0.75055299$, and $y_2 = 0.64699237$ at the fifth iteration with the error tolerance of 10^{-6}. The required digital compensator is

$$G_c(z) = \frac{11.869083z^2 - 13.49237z + 3.0584008}{z^2 - 0.75055299z + 0.64699237}$$

$$= \frac{11.869083(z - 0.82408067)(z - 0.31268533)}{(z - 0.37527650 + j0.71144917)(z - 0.37527650 - j0.71144917)} \qquad (21a)$$

The Nyquist plot of $G_{e1}(z) \triangleq G_c(z)G_nG_o(z)$ is shown in Fig. 2. It closely matches the Nyquist plot of $G_e(s)$. The closed-loop pulse-transfer function is

$$T_{e1}(z) = \frac{\begin{aligned}&0.4861004207\times10^{-3}z^8 + 0.2445680553\times10^{-2}z^7 - 0.6339988815\times10^{-2}z^6 \\ &+ 0.2419157047\times10^{-2}z^5 + 0.2591699688\times10^{-2}z^4 - 0.1845418962\times10^{-2}z \\ &+ 0.2163673922\times10^{-3}z^2 + 0.2694091451\times10^{-4}z - 0.4154160183\times10^{-9}\end{aligned}}{\begin{aligned}&z^9 - 4.870067017z^8 + 10.63662758z^7 - 13.9112306z^6 + 12.03802088z \\ &- 7.023068435z^4 + 2.678803581z^3 - 0.6134429209z^2 \\ &+ 0.6435845178\times10^{-1}z - 0.9921078960\times10^{-6}\end{aligned}} \qquad (21b)$$

The unit-step responses of the existing stabilized continuous-data system $T_e(s)$ in Eq. (6) and $T_{e1}(z)$ in Eq. (21b) are shown for comparison in Fig. 3. The time-response of the newly designed sample data system is very close to the existing stabilized system. It is interesting to note that $G_c(s)$ of Eq. (5b) is a fourth-order analog controller whereas the $G_c(z)$ of Eq. (21a) is a second-order digital controller.

In a large control system it is often difficult to select a minimum common sampling period among the various subsystems. For example, the missile inner loop stability system with sampling period T = 0.008 sec is used with a terminal guidance system. The terminal guidance system is low-pass. Therefore, a larger sampling period may be economically used in this system. If we assign a larger sampling period T_1(=0.015 sec) for the outer guidance loop, and we desire a single sample period, we must raise the sampling period T(=0.008 sec) of the actuator and inner loop from T(=0.008) to T_1(=0.015). Notice that the new sampling frequency ω_{s1} $(=2\pi/T_1 = 418.88) \gg 2\omega_{140}$(=280 rad/sec). The modified open-loop pulse-transfer function with $T_1 = 0.015$ sec is

$$G_nG_o^*(z) = \frac{a_0z^6 + a_1z^5 + \ldots + a_6}{b_0z^7 + b_1z^6 + \ldots + b_7} \tag{22}$$

where

$a_0 = 0.3733134407 \times 10^{-3}$ $b_0 = 1.0$

$a_1 = 0.1570750710 \times 10^{-2}$ $b_1 = -3.257024652$

$a_2 = -0.169231262 \times 10^{-2}$ $b_2 = 3.855486034$

$a_3 = -0.5827024448 \times 10^{-3}$ $b_3 = -2.039756267$

$a_4 = 0.3184265948 \times 10^{-3}$ $b_4 = 0.5526488259$

$a_5 = 0.1895463329 \times 10^{-4}$ $b_5 = -0.1245437594$

$a_6 = 0.11354062 \times 10^{-9}$ $b_6 = 0.1318981958 \times 10^{-1}$

$b_7 = -0.1252480548 \times 10^{-10}$

Since a larger sampling period T_1 is used, we select a third order digital controller $G_c^*(z)$ rather than the second order digital controller. $G_c(z)$ of Eq. (21a) is

$$G_c^*(z) = \frac{x_0 z^3 + x_1 z^2 + x_2 z + x_3}{z^3 + y_1 z^2 + y_2 z + y_3} \qquad (23)$$

The x_ℓ and y_ℓ are seven unknown constants to be determined. $G_e(j\omega)$ at $\omega = 0 \triangleq \omega_0$, $\omega = 1.9 = \omega_{1.9}$, $\omega = 3.2 \triangleq \omega_{3.2}$, and $\omega = 140 \triangleq \omega_{140}$ shown in Eq. (7) are used as the dominant data to determine x_ℓ and y_ℓ. Using the above design procedure, we can determine a set of equations $f_i(x_\ell, y_\ell) = 0$, $i = 1,2,\ldots,7$. These equations can be solved by using the Newton-Raphson method, with the set of initial estimates obtained from Eq. (15c). The data obtained from Eq. (18) at $\omega = 0.01 \triangleq \omega_{0.01}$, $\omega_{1.9}$, $\omega_{3.2}$, and ω_{140} are used in Eq. (15c) to determine the initial estimates $x_0^* = 13.177031$, $x_1^* = -25.535836$, $x_2^* = 14.643983$, $x_3^* = -2.2787512$, $y_1^* = -0.37332775$, $y_2^* = -0.32614954$, and $y_3^* = -0.296499004$. Using these values as initial values for the Newton-Raphson method, gives the desired constants x_ℓ and y_ℓ at the 17th iteration with error tolerance of 10^{-6}. The newly designed digital controller $G_c^*(z)$ is

$$G_c^*(z) = \frac{13.170704z^3 - 25.531430z^2 + 14.629635z - 2.2685451}{z^3 - 0.37424841z^2 - 0.32757047z - 0.29794756} \qquad (24)$$

The closed-loop pulse-transfer function is

$$\frac{Y(z)}{R(z)} \triangleq T_{e2}(z) = \frac{G_{e2}(z)}{1 + G_{e2}(z)} = \frac{b_0 z^9 + b_1 z^8 + \ldots + b_9}{a_0 z^{10} + a_1 z^9 + \ldots + a_{10}} \qquad (25)$$

where

$G_{e2}(z) = G_c^*(z) G_n G_o^*(z)$

$a_0 = 1.0$	$b_0 = 0.4916800827 \times 10^{-2}$
$a_1 = -3.626356261$	$b_1 = 0.1115666668 \times 10^{-1}$
$a_2 = 4.758008528$	$b_2 = -0.5693102109 \times 10^{-1}$

$a_3 = -2.77063927$ | $b_3 = 0.5766519111 \times 10^{-1}$

$a_4 = 1.081168732$ | $b_4 = -0.9250105745 \times 10^{-2}$

$a_5 = -0.8211905469$ | $b_5 = -0.1256587702 \times 10^{-1}$

$a_6 = 0.4739432136$ | $b_6 = 0.5496414235 \times 10^{-2}$

$a_7 = -0.1233033662$ | $b_7 = -0.4450686236 \times 10^{-3}$

$a_8 = 0.3234184521 \times 10^{-1}$ | $b_8 = -0.4299777941 \times 10^{-4}$

$a_9 = -0.3972872336 \times 10^{-2}$ | $b_9 = -0.2575720172 \times 10^{-9}$

$a_{10} = -0.253840284 \times 10^{-9}$

The Nyquist plot of $G_{e2}(z)$ shown in Fig. 2 matches very well that of $G_e(s)$. The unit-step response for $Y(z)$ in Eq. (25) is shown in Fig. 3. The time response of $T_{e2}(z)$ very closely matches that of the original system $T_e(s)$. The resulting design is seen to be quite satisfactory.

V. CONCLUSION

A dominant-data matching method has been given for fitting the coefficients of a pulse-transfer function from available time and frequency response data or from assigned design goals expressed by a set of control specifications. When the dominant data are obtained from a high-order continuous-data as well as a discrete-data system, our new method has been used to determine a reduced-order discrete-data system. If the data are experimental time and frequency response data of a system to be identified, our method may be used to identify the pulse-transfer function. Also, the method has been used for redesigns of a continuous system using a digital filter with various sampling periods. The pulse-transfer function obtained by our new method has the exact dominant performance of the original or desired system. We feel that the flexibility and accuracy of our new method will have significant practical advantages for the design of digital systems.

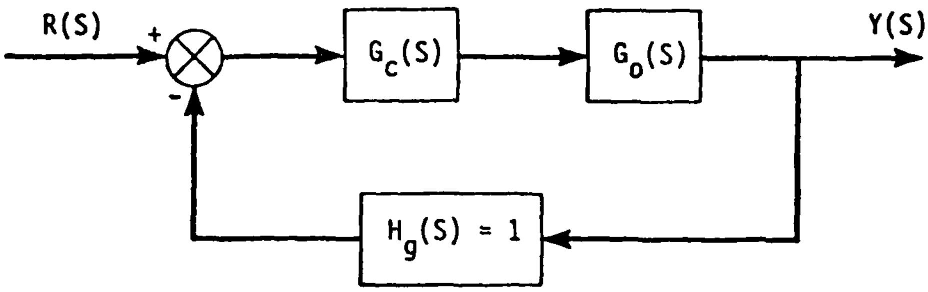


Figure 1. Block Diagram of a Missile Pitch Control System

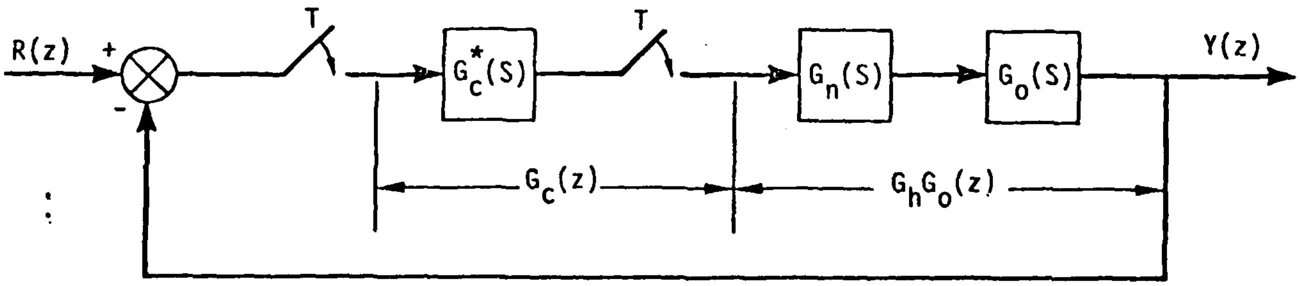


Figure 4. Block Diagram of a Digital Pitch Control System

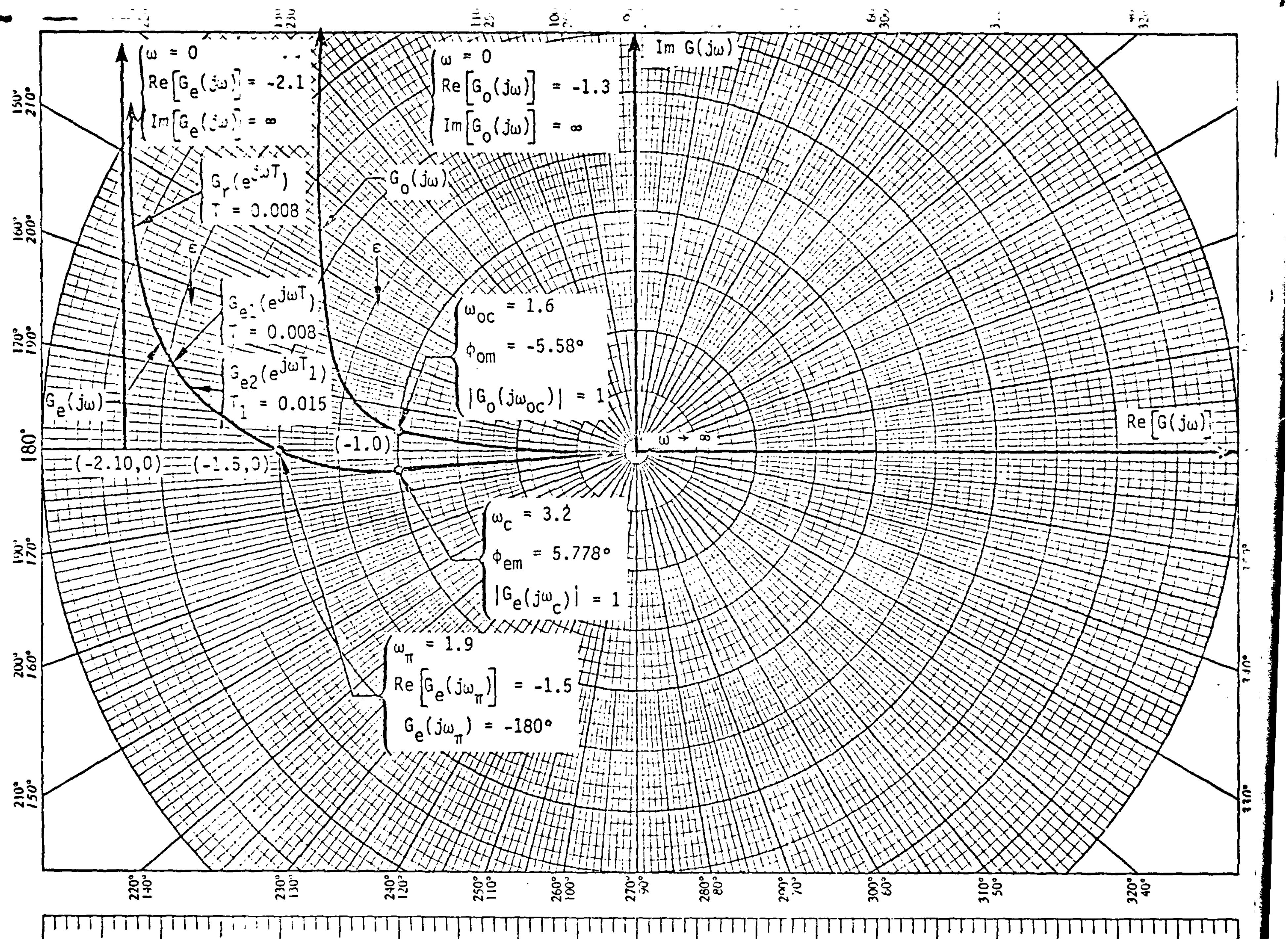


Figure 2. The Nyquist Plots of Various Example Open-Loop Transfer Functions

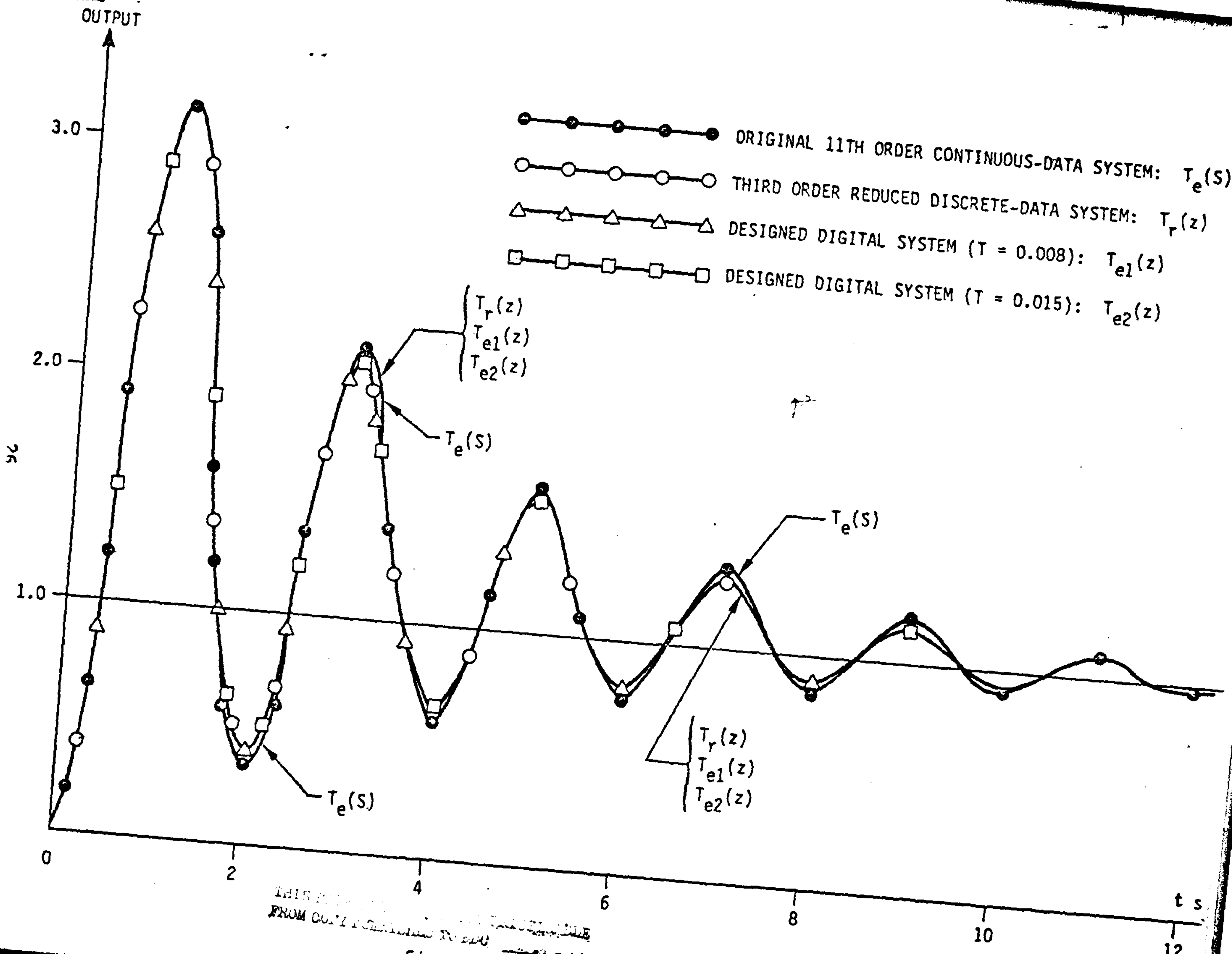


Figure 3. Time-Responses of the Various Systems

Acknowledgments

This work was supported in part by U.S. Army Research Office, DAAG 29-79-C-0178, and U.S. Army Missile Research and Development Command, DAAK 40-79-C-0061.

REFERENCES

[1] E. I. Jury, Sampled-Data Control Systems, Wiley, New York, 1958.

[2] J. T. Tou, Digital and Sampled-Data Control Systems, McGraw-Hill Co., New York, 1959.

[3] J. R. Ragazzini and G. F. Franklin, Sampled-Data Control Systems, McGraw-Hill Co., New York, 1958.

[4] B. C. Kuo, Analysis and Synthesis of Sampled-Data Control Systems, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1963.

[5] B. C. Kuo, Digital Control Systems, SRL Publishing Co., Illinois, 1977.

[6] K. S. Rattan and H. H. Yeh, "Discretizing Continuous-Data Control Systems," Computer-Aided Design, Vol. 10, No. 5, pp. 299-306, Sept. 1978.

[7] E. C. Levy, "Complex Curve Fitting," IRE Trans. on Automatic Control, Vol. AC-4, pp. 37-44, May 1959.

[8] C. K. Sanathanan and J. Koerner, "Transfer Function Synthesis as a Ratio of Two Complex Polynomials," IEEE Trans. on Automatic Control, Vol. AC-8, pp. 56-58, 1963.

[9] J. E. Gibson and Z. V. Rekasius, "A Set of Standard Specifications for Linear Automatic Control Systems," AIEE Trans. Application and Industry, pp. 65-77, May 1961.

[10] G. S. Axelby, "Practical Methods of Determining Feedback Control Loop Performance," Proc. Ist IFAC, pp. 68-74, 1960.

[11] J. G. Truxal, Control System Synthesis, McGraw-Hill Co., New York, pp. 76-87, 1955.

[12] L. S. Shieh, Y. J. Wei, H. Z. Chow, and R. E. Yates, "Determination of Equivalent Dominant Poles and Zeros Using Industrial Specifications," to be published in IEEE Trans. on IECI, 1979.

[13] G. D. Byrne and C. A. Hall, Numerical Solution of Systems of Nonlinear Algebraic Equations, Academic Press, New York, 1973.

[14] IBM S/370-360 Reference Manual IMSL (The International Mathematical and Statistical Library).

[15] J. T. Bosley, "Digital Realization of the T-6 Missile Analog Autopilot," Final Report, DAAK40-77-C-0048 TGT-001, May 1977.

[16] K. W. Plunkett, J. Meadows, W. Albanes, and J. Bosley, "Design and Analysis of a Microprocessor-Based Digital Autopilot for Terminal Homing Missiles," Final Report, T-78-57, U.S. Army Missile R&D Command, Alabama, May 1978.

Chapter III

A MODIFIED DIRECT-DECOUPLING METHOD FOR MULTIVARIABLE CONTROL SYSTEM DESIGNS

L. S. Shieh[1], Y. J. Wei[1], and R. E. Yates[2]

ABSTRACT

A design method, which decouples an interactive system by using a compensator obtained from the plant inverse matrix, which is often called the direct-decoupling method is modified in this paper. The modified direct-decoupling method uses the adjoint matrix instead of the inverse of the plant matrix to construct the compensator. The method uses a frequency-domain model-reduction method to simplify the degree of the given plant transfer function matrix and the obtained compensator. For an open-loop stable multivariable system, the proposed method gives a simple, practical and realizable controller without using an unstable pole-zero cancellation approach.

[1]L. S. Shieh and Y. J. Wei are with the Department of Electrical Engineering, University of Houston, Houston, Texas 77004.

[2]R. E. Yates is with the Guidance and Control Directorate, U. S. Army Missile Research and Development Command, Redstone Arsenal, Alabama 35809.

I. INTRODUCTION

For the general multi-input-output control system, each input affects several outputs and there are many degrees of freedom for system design. Therefore, it is difficult to control and/or design such a system. The removal of the interactive control effects and the application of the well-developed classical design techniques of a single-variable system to the decoupled system is one popular design method. This method is often called the decoupling method of multivariable system design.

In the time domain, the decoupling problem has been studied via state-space techniques by several authors [1-5]. The conditions for the existence of a decoupling system have been developed in these pioneering works [1-5]. The state-space techniques are concerned with the internal structure of the multivariable system. Thus, the limitations of the decoupling approach can be derived, and a simple state feedback controller can be designed to achieve an optimal result. However, for a real system, many of the states are not accessible. As a result, a high degree observer is often required for practical application of the state-space technique.

In the frequency domain, decoupling via the use of transfer function matrices has been investigated by several pioneers [6-10] as long as twenty years ago. Since then, practicing engineers have successfully extended the classical frequency-domain approach, (for example, the Nyquist method [11-13] and the root-locus method [14-15]) from single-variable system to multi-variable systems. Most existing frequency-domain design methods for multivariable systems either neglect the effects of weakly interacting subsystems or completely destroy the coupling effects of the original multivariable system such that a simple classical single-variable method can be applied to achieve reasonable design goals. The approach that removes the interactions of the coupled system and designs a controller for each decoupled system by using a compensator obtained from the plant inverse matrix is called the direct-decoupling method. This method is straight forward but the

following problems [16] arise:

a. The existence of the plant inverse matrix

b. The realizability of the obtained high-degree controller

c. The stability of the designed system when unstable pole-zero cancellation has been used

d. The design procedures for a high-degree coupled system is complex.

Since many practical systems [12,13,15,17] are invertible, controllable, observable, and open-loop stable, the direct-decoupling method is worthy of improvement and modification for practical application. Recently, Peczkowski and Sain [17] have improved the method via a time-domain model reduction method and have successfully applied it to design the control system for a F-100 engine. The direct-decoupling method is modified and improved in this paper. Our method uses the adjoint matrix instead of the inverse matrix of the plant matrix, and utilizes a frequency-domain model-reduction method for a class of multivariable control system design.

II. MODEL REDUCTION OF A MULTIVARIABLE SYSTEM

The usual procedures for designing high-order linearized multivariable systems are cumbersome and computationally difficult. The controller obtained will usually be a high-degree dynamic controller. To overcome these difficulties, a reduced-order model of the original system is necessary. Recently, various model reduction methods in the frequency domain [18-21] have been proposed for determination of simplified models or for estimation of location of the approximate dominant poles and zeros of the original systems. Since each linearized model or reduced-order model is only an approximate model of the actual system, we may use a reduced-order model as a reference for the control system design. Thus, the design procedure is greatly simplified, and a satisfactory lower-degree controller can be obtained. In this paper, various mixed methods will be given

for obtaining the reduced-order model of the original multivariable control system. The mixed methods are described in the following paragraphs.

Consider a typical transfer function matrix of a multivariable control system

$$G_o(s) = \frac{1}{d_o(s)} \Phi(s) = \frac{1}{d_o(s)}\left\{\Phi_{i,j}(s)\right\} = \left\{g_{i,j}(s)\right\} \tag{1}$$

where each polynomial $\Phi_{i,j}(s)$ is the element at the ith-row and jth-column in $\Phi(s)$, and $d_o(s)$ is the least common denominator of the elements of $\Phi(s)$. The transfer function $g_{i,j}(s)$, which is an element in $G_o(s)$, can be expanded into a continued fraction of the Cauer second form [18] by using repeated long division of the two polynomials to determine the various reduced-order models, that is

$$g_{i,j}(s) = \frac{\Phi_{i,j}(s)}{d_o(s)} = \frac{(b_o+b_1s+\ldots+b_ms^m)_{i,j}}{a_o+a_1s+\ldots+a_ns^n} \tag{2a}$$

$$= \left.\frac{1}{h_1 + \cfrac{s}{h_2 + \cfrac{s}{\ddots}}}\right|_{i,j} \tag{2b}$$

$$\cong \left.\frac{1}{h_1 + \cfrac{s}{h_2}}\right|_{i,j} = \left.\frac{h_2}{h_1h_2+s}\right|_{i,j} \tag{2c}$$

$$\cong \left.\frac{1}{h_1 + \cfrac{s}{h_2 + \cfrac{s}{h_3 + \cfrac{s}{h_4}}}}\right|_{i,j} = \left.\frac{h_2h_3h_4+(h_2+h_4)s}{h_1h_2h_3h_4+(h_1h_2+h_1h_4+h_3h_4)s+s^2}\right|_{i,j}$$

$$\cong \ldots \tag{2d}$$

It has been shown that the first several h's are the dominant quotients in the expansion if the steady-state response is of interest. Bosley et al. [22] have linked the quotients h_i and the time moments (or the moments) of the original system. The reduced-order models in Eq.(2) give good approximate steady-state responses. The disadvantages of the above method is that the reduced-degree models may not be stable even when the original system is stable. To obtain the stable reduced-degree model of each element in $G_o(s)$ and to have the same least common denominator polynomial in the reduced-degree multivariable system, we may use the following mixed methods.

Let the reduced model of the original system in Eq. (1) be

$$G_o^*(s) = \frac{1}{d_o^*(s)} \Phi^*(s) = \frac{1}{d_o^*(s)}\left\{\Phi^*_{i,j}(s)\right\} = \left\{g_{i,j}^*(s)\right\} \tag{3a}$$

where

$$g^*_{i,j}(s) = \frac{\Phi^*_{i,j}(s)}{d_o^*(s)} = \frac{\left\{b_o^*+b_1^*s+\ldots+b_{p-1}^*s^{p-1}\right\}_{i,j}}{a_o^*+a_1^*s+\ldots+a_p^*s^p}, \quad a_p^* = 1 \tag{3b}$$

The relationships of the dominant quotients h_i in Eq. (2) and the coefficients a_i^* and b_i^* in Eq. (3b) can be expressed in the following matrix equation [23] by

$$[b] = [H][a] \tag{4}$$

where

$$[a]^T = [a_o^*, a_1^*, \ldots, a_{p-1}^*]$$

$$[b]^T = [b_o^*, b_1^*, \ldots, b_{p-1}^*]$$

$$[H] = [H_2]^{-1}[H_1]$$

where T designates transpose, and

$$[H_2] = \begin{bmatrix} h_1 & 0 & 0 & . & 0 & 0 \\ 1 & h_2 & 0 & . & 0 & 0 \\ 0 & 1 & h_3 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_p \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & h_1 & 0 & . & 0 & 0 \\ 0 & 1 & h_2 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_{p-1} \end{bmatrix} \cdots \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & 1 & 0 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ 0 & 0 & 0 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 0 & h_1 \end{bmatrix}$$

$$[H_1] = \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & h_2 & 0 & . & 0 & 0 \\ 0 & 1 & h_3 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_p \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & 1 & 0 & . & 0 & 0 \\ 0 & 0 & h_2 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_{p-1} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & 1 & 0 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ 0 & 0 & 0 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 0 & h_2 \end{bmatrix}$$

The h_i are obtained from Eq. (2) and a_i^* can be determined from the dominant poles of $d_o(s)$ [21], or from the Routh table suggested by Hutton and Friedland [20], or the Routh table of Gustafson [24]. Once h_i and a_i^* are known, the b_i^* can be determined from Eq. (4). Thus the polynomial $\Phi_{i,j}^*(s)$ can be determined, and the equivalent dominant zeros with the preassigned dominant poles obtained from $d_o^*(s)$ can also be determined. Since each subsystem in the reduced-order model has the dominant quotients h_i (or equivalent moments) and the dominant poles of the subsystem in the original system, the reduced-order model closely approximates the transient and steady-state responses of the original system. It also has the same tracking properties of the original system. Thus, the reduced-order model $G_o^*(s)$ is a good approximation of the original system

$G_o(s)$. Using $G_o^*(s)$ as a design reference model, the design process can be greatly simplified and a satisfactory low-degree compensator obtained. This designed system retains the equivalent transmission zeros [25-27] of the original system. Our method is described in the following section.

III. A MODIFIED DIRECT-DECOUPLING METHOD

Let the open-loop transfer function matrix $G_o(s)$ of a unit-matrix feedback multivariable system be

$$G_o(s) = \frac{1}{d_o(s)} \Phi(s) \tag{5}$$

where $G_o(s) \in R(s)^{mxm}$ is a proper transfer function matrix and $d_o(s) \in R(s)$ is the least common denominator polynomial of $G_o(s)$ with degree n. The expression $\Phi(s) \in R[s]^{mxm}$ is the numerator matrix polynomial. Applying the first cascade precompensator $K_1(s)$ to the $G_o(s)$ yields

$$Q_1(s) = G_o(s)K_1(s) = \frac{1}{d_o(s)} \Phi(s)K_1(s) = \text{diag}\left\{\frac{p(s)}{d_o(s)}\right\} \tag{6a}$$

where

$$K_1(s) = \text{adj } \Phi(s) \tag{6b}$$

and

$$P(s) = \det \Phi(s) \tag{6c}$$

Note that p(s) is a numerator polynomial in Eq. (6a). If some zeros of p(s) are in the right half-plane and all zeros of $d_o(s)$ in the left half-plane, a practical compensator can be designed without using unstable pole-zero cancellation. On the other hand, if the direct-decoupling method is applied, the p(s) will be a denominator polynomial in Eq. (6a). Thus the decoupled system is unstable and the impractical pole-zero cancellation

approach would be required to stabilize the system. Thus, an obvious advantage of our method over the old method. Employing the second cascade compensator $K_2(s)$ to $Q_1(s)$, we have the diagonalized open-loop transfer function matrix

$$G_d(s) = Q_1(s)K_2(s) = \text{diag}\left\{\frac{p(s)k_i n_i(s)}{d_o(s)d_i(s)}\right\} \tag{7a}$$

where

$$K_2(s) = \text{diag}\left\{\frac{k_i n_i(s)}{d_i(s)}\right\}, \quad i = 1,2,\ldots,m \tag{7b}$$

Each k_i is an undetermined gain at the ith diagonal element of $G_d(s)$ for the use of the root-locus method. Each $n_i(s)$ and $d_i(s)$ is a scalar polynomial to be assigned in the design process. The assignment of $n_i(s)$ and $d_i(s)$ shall improve the performance of the designed system with the constraint that the cascade compensator $K(s) = K_1(s)K_2(s)$ be realizable. The choice of $n_i(s)$ and $d_i(s)$ is a design freedom and experience is helpful. The total compensator becomes

$$K(s) = K_1(s)K_2(s) = \text{adj}\ \Phi(s)\text{diag}\left\{\frac{k_i n_i(s)}{d_i(s)}\right\} \tag{8}$$

Notice that $G_d(s)$ in Eq. (7) retains some of the transmission zeros of the original system. This can be shown as follows:

$$\text{Let } G_o(s) = \frac{1}{d_o(s)}\Phi(s) = N_\gamma(s)D_\gamma(s)^{-1} \tag{9}$$

where $N_\gamma(s) \in R[s]^{m\times m}$ and $D_\gamma(s) \in R[s]^{m\times m}$ are a pair of relatively prime matrix polynomials. The characteristic poles of $G_o(s)$ are the zeros of det $D_\gamma(s) = 0$. The transmission zeros of $G_o(s)$ are the zeros of det $N_\gamma(s) = 0$.

Eq. (9) can be written as

$$G_o(s) = \frac{\Phi(s)}{d_o(s)} = \frac{N_\gamma(s)\text{adj } D_\gamma(s)}{\Delta(s)} \tag{10a}$$

where

$$
\begin{aligned}
\Delta(s) &= \det D_\gamma(s) \\
&= \text{the characteristic polynomial of } G_o(s) \\
&= \text{the least common-denominator polynomial of all minors} \\
&\quad \text{of } G_o(s).
\end{aligned} \tag{10b}
$$

Taking the determinants of both sides of Eq. (10a) yields

$$
\begin{aligned}
\det\left[\frac{\Phi(s)}{d_o(s)}\right] &= \frac{p(s)}{d_o{}^m(s)} = \frac{\det N_\gamma(s)\det[\operatorname{adj} D_\gamma(s)]}{\Delta^m(s)} \\
&= \frac{\Delta^{m-1}(s)\det N_\gamma(s)}{\Delta^m(s)} = \frac{\det N_\gamma(s)}{\Delta(s)}
\end{aligned} \tag{11}
$$

Rearranging Eq. (11) gives

$$
\frac{p(s)}{d_o(s)} = \frac{d_o{}^{m-1}(s)}{\Delta(s)} \det N_\gamma(s) \tag{12}
$$

Substituting Eq. (12) into Eq. (7a) we have

$$
G_d(s) = \operatorname{diag}\left\{\frac{k_i n_i(s) d_o{}^{m-1}(s)\det N_\gamma(s)}{d_i(s)\ \Delta(s)}\right\} \tag{13a}
$$

and the closed-loop system is

$$
Y(s) = \operatorname{diag}\left\{\frac{k_i n_i(s) d_o{}^{m-1}(s)\det N_\gamma(s)}{\Delta(s) d_i(s) + k_i d_o{}^{m-1}(s) n_i(s)\det N_\gamma(s)}\right\} R(s) \tag{13b}
$$

where $Y(s)$ and $R(s)$ are the output and input vectors, respectively.

Since the transmission zeroes of the original system $G_o(s)$ in Eq. (5) are the zeros of $\det N_\gamma(s)$ which appear in both $G_d(s)$ in Eq. (13a) and the designed closed-loop system in Eq. (13b), the designed system retains some of the invariant transmission zeros when $\Delta(s)$ and $\det N_\gamma(s)$ have no common factors. In general, there may exist common factors between $\Delta(s)$ and $\det N_\gamma(s)$. Therefore, $G_d(s)$ may retain only a subset of the transmission zeros. For the

special case, when m=2, $\Delta(s) = d_o^2(s)$, $n_i(s) = 1$, and $d_i(s) = d_o(s)$ the compensated system $G_d(s) = (\text{diag}\{k_i \det N_\gamma(s)/\det D_\gamma(s)\})$ in Eq. (13a) has the exact transmission zeros as well as the characteristic poles of the original system $G_o(s)$.

When the original system $G_o(s)$ has a high-degree transfer function matrix, the reduced-order model $G_o^*(s)$ in Eq. (3a) can be used. Thus, the designed controller K(s) is low-degree and the designed system retains a subset of equivalent transmission zeros of the original system. Note that, even if $\Delta(s)$ and $\det N_\gamma(s)$ in Eq. (13a) have common factors, the $\Delta^*(s)$ and $\det N_\gamma^*(s)$ obtained from the reduced-degree model $G^*(s)$ may have no common factor.

IV. ILLUSTRATIVE EXAMPLES

Example 1

To illustrate the design procedure, we use Mueller's two-shaft aircraft gas turbine [28] example. The open-loop transfer function matrix $G_o(s)$ of the example unit-matrix feedback system is

$$G_o(s) = \frac{1}{d_o(s)} \Phi(s) = \frac{1}{d_o(s)} \begin{bmatrix} \phi_{11}(s) & \phi_{12}(s) \\ \phi_{21}(s) & \phi_{22}(s) \end{bmatrix} \tag{14}$$

where

$$\begin{aligned} d_o(s) &= s^4 + 113.225s^3 + 1357.277s^2 + 3502.95s + 2526.85 \\ \phi_{11}(s) &= 14.9s^2 + 1506.472s + 2543.2 \\ \phi_{12}(s) &= 95150s^2 + 1132094.7s + 1805947 \\ \phi_{21}(s) &= 85.2s^2 + 8642.888s + 12268.8 \\ \phi_{22}(s) &= 124000s^2 + 1492588s + 2525880 \end{aligned}$$

Notice that $d_o(s)$, the least common denominator polynomial, is the characteristic polynomial of $G_o(s)$, or $d_o(s) = \Delta(s)$. Also, this system has no finite transmission zeros. McMorran [12]

designed a dynamic compensator for this system using Rosenbrock's inverse Nyquist array method [11]. His design goals are a weakly coupled system and a fast response. The compensator obtained was

$$K_e(s) = \frac{1}{s(s+158.5)}\begin{bmatrix} -26.847s(s+20.6) & 46.272(s+1.706)(s+11.6) \\ 0.018468s(s+146.3) & -0.00726(s+1.706)(s+101.4) \end{bmatrix} \tag{15}$$

Unit-step responses are shown in Figure 1.

The goals of our design are:

(a) Two identical diagonal subsystems decoupled in the closed-loop system

(b) Unity final values of unit-step responses

(c) Less than 10 percent maximum overshoot

(d) The time required for the unit-step response to reach the first peak of the overshoot t_p is 0.01 sec.

For this low-order system, use of the reduced-degree model is not necessary. To simplify the controller in Eq. (8) let $n_i(s) = 1$. Then K(s) becomes

$$K(s) = \text{adj}\ \Phi(s)\ K_2(s) = \begin{bmatrix} \phi_{22}(s) & -\phi_{12}(s) \\ -\phi_{21}(s) & \phi_{11}(s) \end{bmatrix}\begin{bmatrix} \frac{k_1}{d_1(s)} & 0 \\ 0 & \frac{k_2}{d_2(s)} \end{bmatrix} \tag{16a}$$

The designed open-loop system is

$$G_d(s) = \begin{bmatrix} \frac{k_1 p(s)}{d_o(s)d_1(s)} & 0 \\ 0 & \frac{k_2 p(s)}{d_o(s)d_2(s)} \end{bmatrix} \tag{16b}$$

where $p(s) = \det \Phi(s) = -6259180d_o(s)$; k_i and $d_i(s)$ are to be determined. To circumvent a negative p(s) in the open-loop transfer function, $G_d(s)$, we apply a unit-matrix controller K_o to $\Phi(s)$ before using the controller K(s) in Eq. (16a). Our objective is

to interchange entries of $\Phi(s)$ from the first column to the second column such that the modified open-loop system $\Phi_m(s)$ becomes

$$\Phi_m(s) = \Phi(s)K_o = \begin{bmatrix} \phi_{11}(s) & \phi_{12}(s) \\ \phi_{21}(s) & \phi_{22}(s) \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} \phi_{12}(s) & \phi_{11}(s) \\ \phi_{22}(s) & \phi_{21}(s) \end{bmatrix} \tag{16c}$$

Thus, the controller for $\Phi_m(s)$ is

$$K_3(s) = \text{adj } \Phi_m(s)K_2(s) = \begin{bmatrix} \phi_{21}(s) & -\phi_{11}(s) \\ -\phi_{22}(s) & \phi_{12}(s) \end{bmatrix} \begin{bmatrix} \frac{k_1}{d_1(s)} & 0 \\ 0 & \frac{k_2}{d_2(s)} \end{bmatrix} \tag{16d}$$

and the modified controller for the original system $G_o(s)$ is

$$K_m(s) = K_o\text{adj } \Phi_m(s)K_2(s) = \begin{bmatrix} -\phi_{22}(s) & \phi_{12}(s) \\ \phi_{21}(s) & -\phi_{11}(s) \end{bmatrix} \begin{bmatrix} \frac{k_1}{d_1(s)} & 0 \\ 0 & \frac{k_2}{d_2(s)} \end{bmatrix} \tag{16e}$$

The designed open-loop system is

$$G_{dm}(s) = \begin{bmatrix} \frac{-k_1 p(s)}{d_o(s)d_1(s)} & 0 \\ 0 & -\frac{k_2 p(s)}{d_o(s)d_2(s)} \end{bmatrix} = \begin{bmatrix} \frac{6259180k_1}{d_1(s)} & 0 \\ 0 & \frac{6259180k_2}{d_2(s)} \end{bmatrix} \tag{16f}$$

To satisfy the first specification, we let $k_1 = k_2 = k$ and $d_1(s) = d_2(s) = d(s)$, thereby reducing the multivariable design to a scalar design problem. In other words, we design $g_d(s) = 6259180k/d(s)$, a diagonal entry in $G_{dm}(s)$, to satisfy other assigned specifications. To meet the requirement of unit-final value, we choose $g_d(s)$ to be "Type 1"; and to meet the other two conditions, overshoot and the peak time, we choose a second-order compensator

$$g_d(s) = \frac{6259180k}{s(s+c)} \tag{17a}$$

The characteristic polynomial of the designed closed-loop system is

$$\Delta_d(s) = s^2 + cs + 6259180k = s^2 + 2\xi\omega_n s + \omega_n^2 \tag{17b}$$

where $c = 2\xi\omega_n$ and $\omega_n^2 = 6259180k$. The parameters ξ (the damping ratio) and ω_n (the undamped natural frequency) are to be determined. From a design rule of thumb [29] we can estimate ω_n knowing t_p

$$\omega_n \cong \frac{\pi}{t_p} \cong \frac{3.14}{0.01} \cong 300 \text{ rad/sec} \tag{17c}$$

Also, another rule of thumb [29] gives

$$M_p \cong e^{-\xi\pi}$$

or

$$\xi \cong -\frac{\ln M_p}{\pi} = -\frac{\ln 0.1}{3.14} \cong 0.75 \tag{17d}$$

Substituting ξ and ω_n into Eq. (17b) we can solve for c and k;

$$c = 2\xi\omega_n = 450$$

and

$$k = 0.01438 \tag{17e}$$

The characteristic polynomial $\Delta_d(s)$ and its poles are

$$\Delta_d(s) = s^2 + 450s + 90000$$

and

$$s_{1,2} = -\xi\omega_n \pm j\omega_n\sqrt{1-\xi^2} = -225 \pm j198.43 \tag{17f}$$

The compensator in Eq. (16e) becomes

$$K_m(s) = \frac{1}{s(s+450)}\begin{bmatrix} -1782.98s^2-21461.74s-36319.33 & 1368.15s^2+16278.3s+25967.5 \\ 1.22508s^2+124.275s+176.4116 & -0.214245s^2-21.66138s-36.5684 \end{bmatrix} \tag{18a}$$

The decoupled closed-loop system is

$$Y(s) = \text{diag}\left\{\frac{90000}{s^2 + 450s + 90000}\right\} R(s) \tag{18b}$$

The unit-step responses of the designed system are shown in Figure 1. The final values of the unit-step responses are unity, and the maximum percentage overshoot is 1 percent. Also, the peak time t_p is 0.014 sec. Note that the designed system and the original system have no finite transmission zeros. Comparing the proposed compensator $K_m(s)$ in Eq. (18a) and that of McMorran in Eq. (15) we observe that both controllers are second order. However, our controller satisfies more sophisticated control specifications than McMorran's. The unit-step response curves show that our design gives less overshoot and less oscillations and is completely decoupled.

Example 2.

To illustrate our design procedure using a reduced-order model, we use a paper-making machine [13] example. The open-loop transfer function matrix $G_o(s)$ of the unit-matrix feedback system is

$$G_o(s) = \frac{1}{d_o(s)} \Phi(s) = \frac{1}{sd_1(s)} \begin{bmatrix} \phi_{11}(s) & \phi_{12}(s) \\ \phi_{21}(s) & \phi_{22}(s) \end{bmatrix} \tag{19}$$

where

$$\begin{aligned} d_1(s) &= s^6 + 34.9798s^5 + 565.584s^4 + 5016.37s^3 + 24517.51s^2 \\ &\quad + 55613.33s + 12868.37 \\ \phi_{11}(s) &= -9.72727s(s^2 + 15.39326s + 112.3596) \\ \phi_{12}(s) &= 173.386s(s^2 + 11.44444s + 55.55556) \\ \phi_{21}(s) &= 0.204545(s^2 + 15.39326s + 112.3596) \\ \phi_{22}(s) &= 19.0s(s^2 + 11.44444s + 55.55556) \end{aligned}$$

Sinha and Rutherford [13] have used Rosenbrock's inverse Nyquist array technique [11] to design a controller for this system. The precontroller to $G_o(s)$ is

$$K_p(s) = \frac{1}{s} \begin{bmatrix} -15(s+1) & 52(s+1.11111) \\ 0 & 3.75(s+1.11111) \end{bmatrix} \tag{20}$$

The unit-step responses of the closed-loop system using $K_p(s)$ are shown in Figure 2.

In this problem, our design goals are:

(a) The designed closed-loop system must be weakly decoupled and have two approximately identical diagonal subsystems

(b) The final values of the unit-step responses must be unity

(c) The tracking error should decay at least as fast as e^{-10t}.

$G_o(s)$ is a high degree transfer-function matrix. To determine a low-degree controller and to simplify the design procedures, a reduced-order model will be used. The continued fraction mixed method [18] and the Routh approximation method [20] may be used to determine the reduced-order model $G_o^*(s)$. The procedures to apply the mixed method are as follows: First, the reciprocal polynomial with respect to $d_1(s)$ in Eq. (19) (that is, $d_1'(s) = \frac{1}{s}d_1(\frac{1}{s})$) is arranged into the Routh array [20] to determine α_1. The α_1 obtained is equal to 0.23139. Thus, the equivalent least common denominator polynomial $d_o^*(s)$ of the $G_o^*(s)$ is

$$d_o^*(s) = s(s+\alpha_1) = s(s+0.23139) \tag{21}$$

Then, various first dominant quotients (defined as $\{h_1\}_{i,j}$) of each entry in $G_o(s)$ can be determined by performing the continued fraction expansion on each $\phi_{i,j}(s)$ and $d_o(s)$ as shown in Eq. (2). These values are

$$\{h_1\}_{1,1} = -11.68221$$

$$\{h_1\}_{1,2} = 1.31079$$

$$\{h_1\}_{2,2} = 11.962 \tag{22a}$$

and the $\{h_1\}_{2,1}$ for $s\phi_{21}(s)$ and $d_o(s)$ is

$$\{h_1\}_{2,1} = 555.5556 \tag{22b}$$

Substituting the α_1 value ($\overset{\Delta}{=} a_o^*$) in Eq. (21) and each $\{h_1\}_{i,j}$ in Eqs. (22) into Eq. (4) yields each reduced-order polynomial $\phi_{i,j}^*(s)$. Thus,

$$G_o^*(s) = \frac{1}{d_o^*(s)}\Phi^*(s) = \frac{1}{d_o^*(s)}\begin{bmatrix} \phi_{11}^*(s) & \phi_{12}^*(s) \\ \phi_{21}^*(s) & \phi_{22}^*(s) \end{bmatrix} \tag{23}$$

where

$$d_o^*(s) = s(s+0.23139)$$

$$\phi_{11}^*(s) = -0.01981s \qquad \phi_{21}^*(s) = 0.0004165$$

$$\phi_{12}^*(s) = 0.17653s \qquad \phi_{22}^*(s) = 0.019344s$$

The unit-step response of $G_o(s)$ and $G_o^*(s)$ are shown in Figure 3. The approximation is excellent. Using $G_o^*(s)$ as a plant and closely following the procedures used in Example 1, we have the modified controller in the form of Eqs. (16). The precontroller is

$$K_m^*(s) = K_o \operatorname{adj} \Phi_m^*(s) K_2(s) = \begin{bmatrix} -\dfrac{k_1 n_1(s)\phi_{22}^*(s)}{d_1(s)} & \dfrac{k_2 n_2(s)\phi_{12}^*(s)}{d_2(s)} \\ \dfrac{k_1 n_1(s)\phi_{21}^*(s)}{d_1(s)} & -\dfrac{k_2 n_2(s)\phi_{11}^*(s)}{d_2(s)} \end{bmatrix} \tag{24}$$

The designed open-loop system is

$$G_{dm}^*(s) = \left\{\operatorname{diag} \frac{-k_i p^*(s) n_i(s)}{d_o^*(s)\ d_i(s)}\right\}$$

$$= \operatorname{diag}\left\{\frac{0.0003832(s+0.191868)k_i n_i(s)}{(s+0.23139)\ d_i(s)}\right\} \text{ for } i = 1,2 \tag{25}$$

where $p^*(s) = \det \Phi^*(s) = -0.0003832s(s+0.191868)$

To simplify the design procedure, we let $n_1(s) = n_2(s) = 1$; and to meet the first design goal we let $k_1 = k_2 = k$ and $d_1(s) = d_2(s) = d(s)$. Thus, we have a single open-loop transfer function

$$g_d(s) = \frac{0.0003832(s+0.191868)k}{(s+0.23139)\ d(s)} \tag{26}$$

From basic root-locus theory [30], we observe that there exists a nearby open-loop pole, ($p_o = -0.23139$) and zero, ($z_o = -0.191868$), in $g_d(s)$. When gain k is increased, a pole of the closed-loop system will migrate from p_o to z_o which is a closed-loop zero also. Thus, the performance of the closed-loop system is heavily dependent on the zeros of d(s). To satisfy the third specification we choose $d(s) = s+10$ such that the tracking error of the designed system will decay at least as fast as e^{-10t}. The choice of gain k is a design freedom which can ensure less influence of the nearby p_o and z_o. In this example, we choose $k = 40$. The overall compensator in Eq. (24) becomes

$$K_m^*(s) = \frac{1}{s+10}\begin{bmatrix} -0.77376s & 7.0612s \\ 0.016667 & 0.7924s \end{bmatrix} \tag{27a}$$

Finally, to achieve the second design goal we add a forward-gain matrix H as shown in Figure 4. The H is

$$H = \begin{bmatrix} \frac{g_d(o)}{1+g_d(o)} & 0 \\ 0 & \frac{g_d(o)}{1+g_d(o)} \end{bmatrix}^{-1} = \begin{bmatrix} 787.786 & 0 \\ 0 & 787.786 \end{bmatrix} \tag{27b}$$

The unit-step responses of the closed-loop designed systems using the compensators $K_m^*(s)$ and H in Eqs. (27) and the reduced-order model $G_o^*(s)$ in Eq. (23) are shown in Figure 2. Also, the unit-step responses of the closed-loop systems using the same compensators in Eqs. (27) and the original system $G_o(s)$ in Eq. (19) are shown in the same figure. The response curves show that the designed system has less overshoot, is less oscillatory, and is nearly completely decoupled. The discrepancy between the response curves of the closed-loop designed systems using $G_o(s)$ and $G_o^*(s)$ appears in the transient regions. This is the result of the model reduction method used. Both curves are seen to be practically decoupled and unit-final values in the steady-state.

V. CONCLUSION

A simple design technique has been given for the design of multivariable control systems. The designed system retains some or all of the invariant transmission zeros of the original system. When the given multivariable control system has a high-degree denominator and high-degree numerator, various mixed methods have been illustrated for determining reduced-order models. By using the obtained reduced-degree model as a reference model, the design procedures are greatly simplified, and a satisfactory low-degree compensator has been designed. The design technique is simple and quite satisfactory as shown in the two illustrative examples. However, when a multivariable control system has a large number of inputs and outputs, the degree of the controller obtained by our method may be large despite using a reduced-order model. When the resulting controller dimension is large, model reduction methods may be utilized to reduce the controller to an acceptable order. Our method is suitable for multivariable systems which have high-degree least common denominator polynomial and low-degree numerator polynomials. Another advantage of our modified method over the original direct-decoupling method is that, for an open-loop stable multivariable system, our method gives a simple, practical and realizable controller without using the unstable pole-zero cancellation approach.

ACKNOWLEGMENTS

This work was supported in part by U. S. Army Research Office, under grant DAAG29-79-C-0178, and U. S. Army Missile Research and Development Command, under contract DAAK-40-79-C-0061.

REFERENCES

[1] B. S. Morgan, "The Synthesis of Linear Multivariable Systems by State-Variable Feedback," IEEE Trans. Automatic Control, Vol. AC-9, No. 4, pp. 405-411, Oct. 1964.

[2] E. G. Gilbert, "Controllability and Observability in Multi-Variable Control Systems," SIAM J. on Control, No. 2, 1963.

[3] P. L. Falb and W. A. Wolovich, "Decoupling in the Design and Synthesis of Multivariable Control Systems," IEEE Trans. Automatic Control, Vol AC-12, No. 6, pp. 651-659, Dec. 1967.

[4] M. L. Sain and J. L. Massey, "Invertibility of Linear Time-Invariant Dynamic Systems," IEEE Trans. Automatic Control, Vol. AC-14, pp. 141-149, Apr. 1969.

[5] A. S. Morse and W. M. Wonhan, "Status of Noninteracting Control, " IEEE Trans. Automatic Control, Vol. AC-16, No. 6, pp. 568-581, Dec. 1971.

[6] M. S. Feder and R. Hood, "Analysis for Control Application of Dynamic Characteristics of Turbojet Engine with Tail-pipe Burning," NACA Technical Note 2183, Washington, D.C. September 1950.

[7] H. Freeman, "A Synthesis Method for Multipole Control Systems," AIEE Trans. on Applications and Industry, Vol. 76, Pt. II, pp. 28-31, March 1957.

[8] R. J. Kavanagh, "Noninteracting Controls in Linear Multivariable Systems," AIEE Trans. on Applications and Industry, Vol. 76, Pt. II, pp. 95-99, May 1957.

[9] H. Freeman, "Stability and Physical Realizeability Considerations in the Synthesis of Multipole Control Systems," AIEE Trans. on Applications and Industry, Vol. 77, Pt. II, pp. 1-5, March 1958.

[10] R. J. Kavanagh, "Multivariable Control System Synthesis," AIEE Trans. on Applications and Industry, Vol. 77, Pt. II, pp. 425-429, Nov. 1958.

[11] H. H. Rosenbrock, "Design of Multivariable Control Systems Using the Inverse Nyquist Array," Proc. IEE, Vol. 116, No. 11, pp. 1929-1936, Nov. 1969.

[12] P. D. McMorran, "Design of Gas-Turbine Controller Using Inverse Nyquist Method," Proc. IEE, Vol. 117, No. 10, pp. 2050-2056, 1970.

[13] A. K. Sinha and D. A. Rutherford, "Computer-Aided Controller Design for Paper Machine," Proc. IEE, Vol. 123, No. 10, pp. 1021-1025, Oct. 1976.

[14] A. G. J. MacFarlane and J. J. Belletrutti, "The Characteristic Locus Design Method," Automatica, Vol. 9, No. 5, pp. 575-588, Sept. 1973.

[15] L. V. Nitsche, L. K. Luessow, and G. J. Thaler, "An Automatic Depth and Pitch Control System for Submarines," in Alternatives for Linear Multivariable Control, National Engineering Consortium Inc., Chicago, pp. 87-96, 1978.

[16] A. J. Fossard, Multivariable System Control, North-Holland Publishing Co., New York, 1977.

[17] J. L. Peczkowski and M. K. Sain, "Linear Multivariable Synthesis with Transfer Functions," in Alternative for Linear Multivariable Control, National Engineering Consortium Inc., Chicago, pp. 111-127, 1978.

[18] C. F. Chen and L. S. Shieh, "A Novel Approach to Linear Model Simplification," Int. J. Control, Vol. 8, No. 6, pp. 561-570, 1968.

[19] Y. Shamash, "Linear Reduction Using Pade Approximation to Allow Retention of Dominant Modes," Int. J. Control, Vol. 21, No. 3, pp. 475-484, 1975.

[20] M. Hutton and B. Friedland, "Routh Approximations for Reducing Order of Linear, Time-Invariant Systems," IEEE Trans. AC-20, No. 3, pp. 329-337, 1975.

[21] L. S. Shieh and Y. Wei, "A Mixed Method for Multivariable System Reductions," IEEE Trans. AC-20, No. 3, pp. 429-433, 1975.

[22] M. J. Bosley, Kropholler, and F. P. Lees, "On the Reduction Between the Continued Fraction Expansion and Moment Matching Methods of Model Reduction," Int. J. Control, Vol. 18, No. 3, pp. 461-474, 1973.

[23] C. F. Chen and L. S. Shieh, "An Algebraic Method for Control Systems Design," Int. J. Control, Vol. 11, No. 5, pp. 717-739, 1970.

[24] R. D. Gustafson, "A Paper and Pencil Control System Design Technique," JACC, pp. 301-310, June 1965.

[25] C. A. Desoer and J. D. Schulman, "Zeros and Poles of Matrix Transfer Functions and Their Dynamical Interpretation," IEEE Trans. Circuits and Systems, Vol. CAS-21, pp. 3-8, Jan. 1974.

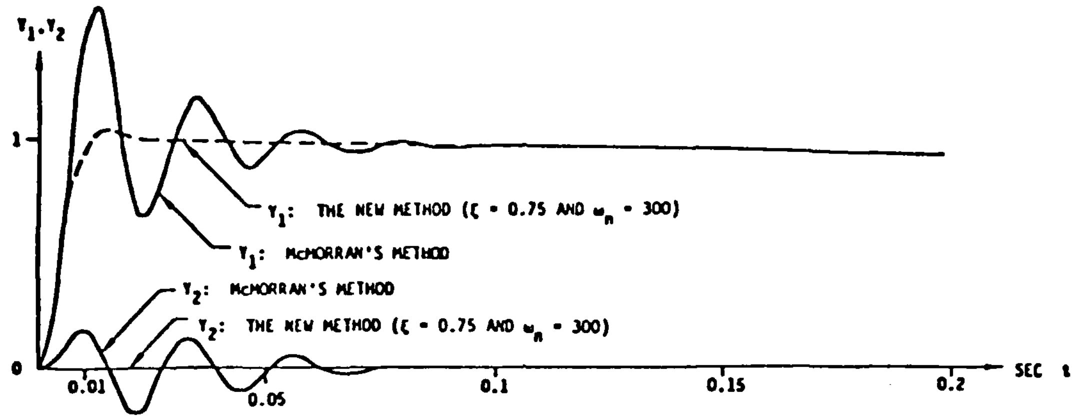
[26] H. Kwakernaak and R. Sivan, Linear Optimal Control Systems, Wiley-Interscience, New York, 1972.

[27] L. S. Shieh, Y. J. Wei, and J. M. Navarro, "An Algebraic Method to Determine the Common Divisor, Poles and Transmission Zeros of Matrix Transfer Functions," Int. J. Systems Science, Vol. 9, No. 9, pp. 949-964, 1978.

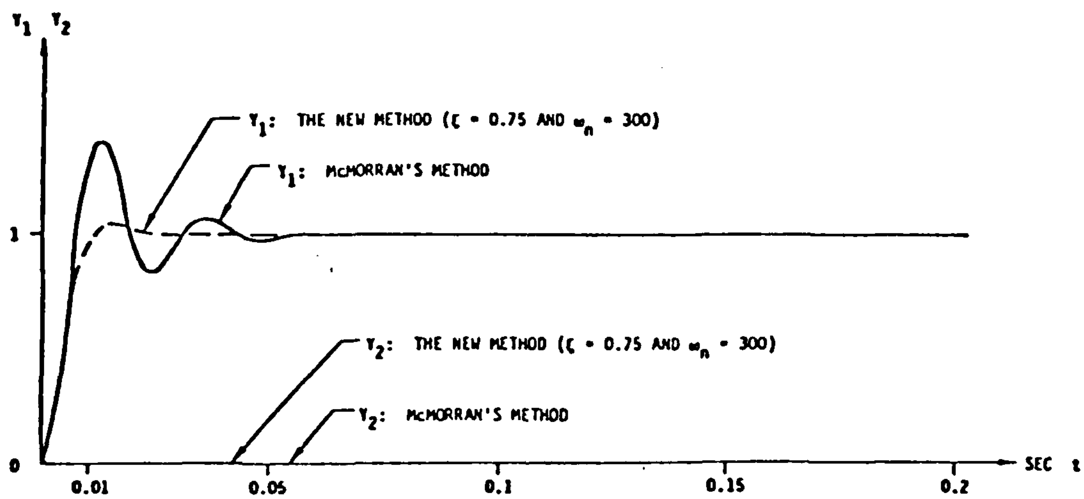
[28] G. S. Mueller, "Linear Model of a Two Shaft Turbo Jet and Its Properties," Proc. IEE, Vol. 118, No. 6, pp. 813-815, 1971.

[29] K. Ogata, Modern Control Engineering, Prentice-Hall Inc., N.J., pp. 216-258, 1970.

[30] G. J. Thaler, Design of Feedback Systems, Dowden, Hutchinson of Ross, Inc., Stroudsburg, Pennsylvania, 1973.

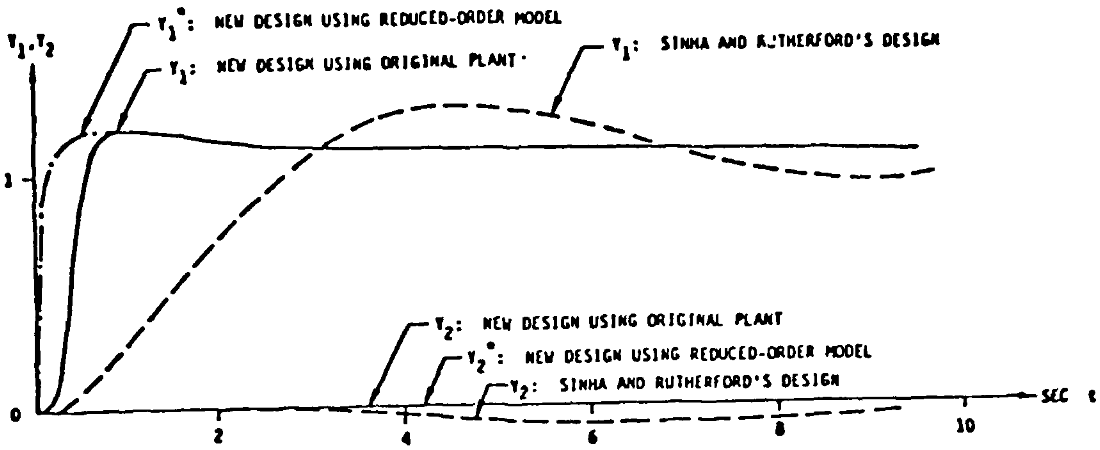


(a) INPUTS: $r_1 = 1$ and $r_2 = 0$

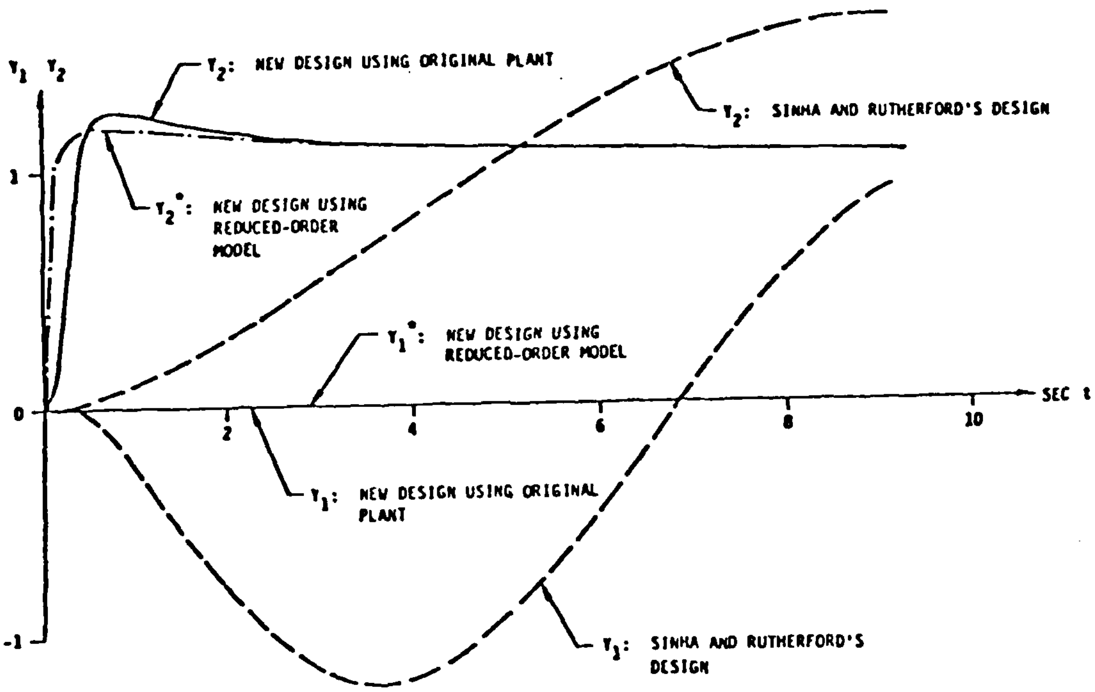


(b) INPUTS: $r_1 = 0$ and $r_2 = 1$

Figure 1. Unit Step Responses of Designed Systems in Example 1

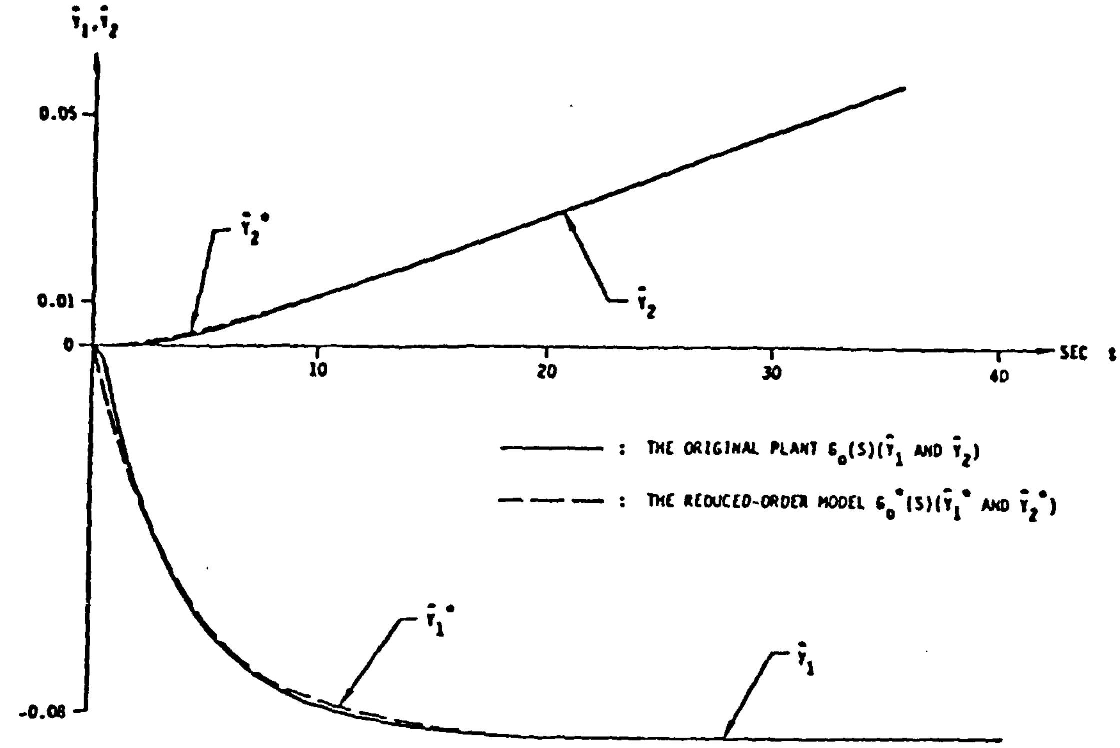


(a) INPUTS: $r_1 = 1$ and $r_2 = 0$

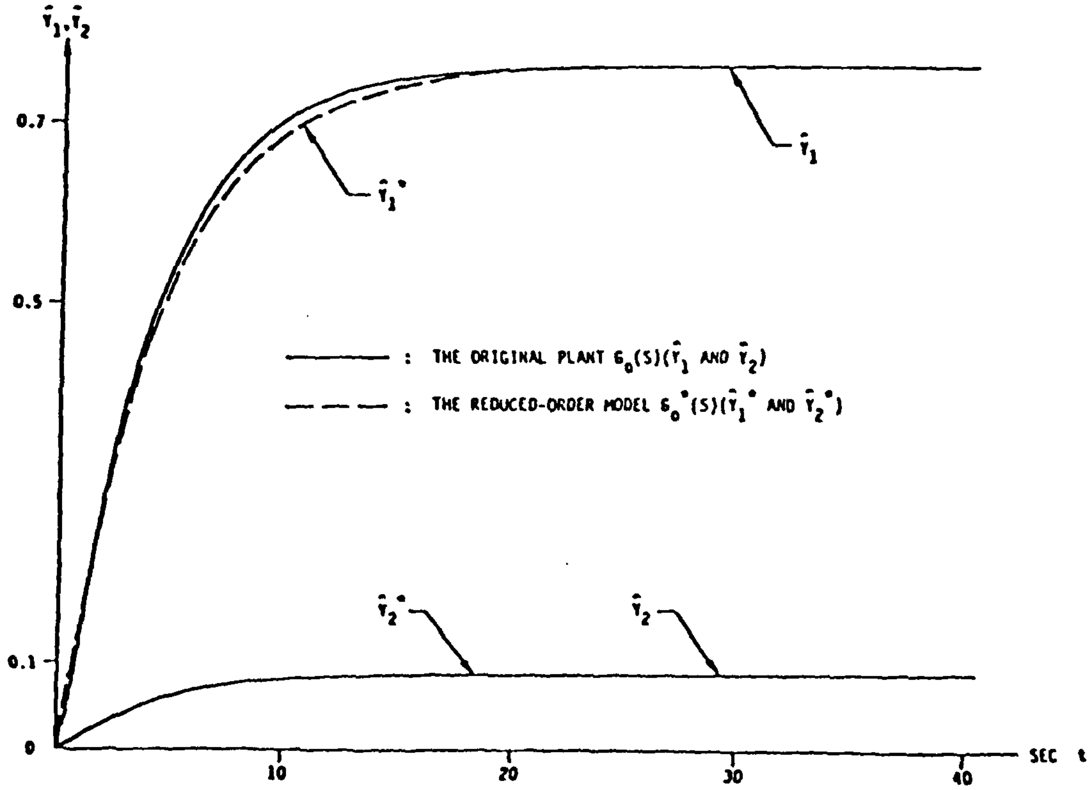


(b) INPUTS: $r_1 = 0$ and $r_2 = 1$

Figure 2. Unit Step Responses of Designed Systems in Example 2



(a) INPUTS: $U_1 = 1$ and $U_2 = 0$



(b) INPUTS: $U_1 = 0$ and $U_2 = 1$

Figure 3. Responses of Original Plant and Reduced-Order Model to a Unit Step in U_1 and U_2

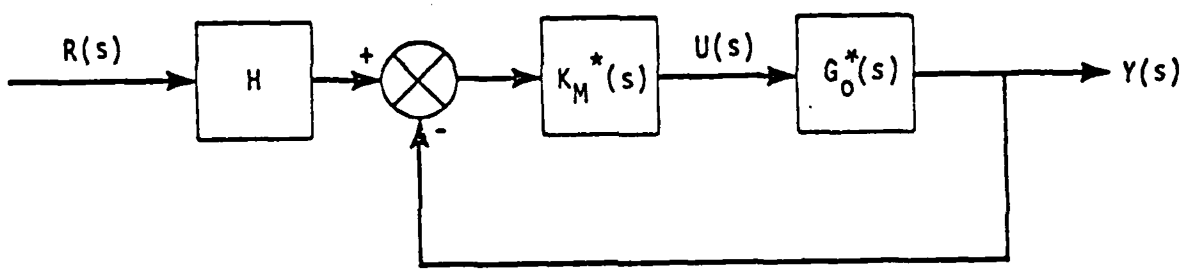


Figure 4. Structure of Designed System

Chapter IV

Some Properties and Applications of a New Matrix Sturm Series and a New Block Canonical Form of a Matrix Transfer Function

L. S. Shieh and A. Tajvari

Department of Electrical Engineering
University of Houston
Houston, Texas 77004

ABSTRACT

An algebraic method is developed to construct a new matrix Sturm series and to establish a new block canonical form of a matrix transfer function. The matrix Sturm sequence is then used to determine the number of real poles of a matrix transfer function that may not consist of a pair of relatively prime matrix polynomials, and it can be used to determine whether an impedance matrix can be realized by using RC elements. The block canonical form is used to construct a new block state equation in the block tridiagonal form and to obtain a pair of relatively left prime matrix polynomials of a matrix transfer function.

1. Introduction

The matrix transfer function T(s) of a multivariable system with an m×1 input vector R(s) and an m×1 output vector y(s) is written as

$$y(s) = T(s)\ R(s) \tag{1a}$$

and

$$T(s) = D_1(s)^{-1} D_2(s) \tag{1b}$$

where

$$D_1(s) = A_{n+1}s^n + A_n s^{n-1} + \ldots + A_2 s + A_1 = D_{11}s^n + D_{12}s^{n-1} + \ldots + D_{1,n}s + D_{1,n+1}$$

$$D_2(s) = B_n s^{n-1} + B_{n-1}s^{n-2} + \ldots + B_2 s + B_1 = D_{21}s^{n-1} + D_{22}s^{n-2} + \ldots + D_{2,n-1}s + D_{2,n}$$

$A_i (=D_{1,j})$ and $B_i (=D_{2,j})$ are m×m matrix coefficients.

In a single-input single-output system, the T(s) in Eq. (1) can be observed as a scalar transfer function or a driving-point impedance function. The T(s) can be formulated into a sequence of polynomials (Sturm sequence) to determine the common polynomial between $D_1(s)$ and $D_2(s)$ and the number of real poles of T(s) by using Sturm's theorem.[1] When no common polynomial between $D_1(s)$ and $D_2(s)$ exists, the T(s) is a function that consists of two relatively prime polynomials, and the dynamic state equation constructed using $D_1(s)$ and $D_2(s)$ is completely controllable and observable. This implies that the dynamic state equation is a minimal realization of T(s). An excellent survey on the applications of the Sturm theorem can be found in

Barnett and Siljak's work.[2]

Recently Bitmead and Anderson[3] have developed the matrix Cauchy theorem and a matrix Sturm series for multivariable systems, and they have discussed their applications to circuit theory. In this paper, an alternate matrix Sturm series is developed and a block canonical form of a matrix transfer function is also derived. Some properties and applications of the developed Sturm series and canonical form to the analyses of multivariable systems are discussed.

II. A Matrix Sturm Series

The $T(s)$ in Eq. (1) is a real rational matrix with left matrix fraction decomposition $D_1(s)^{-1}D_2(s)$. If $D_2(s)$ is non singular, the inversion of $T(s)$ is expressed as

$$T(s)^{-1} = D_2(s)^{-1}D_1(s) = sK_1+D_2(s)^{-1}D_3(s) \tag{2a}$$

or

$$D_1(s) = D_2(s)sK_1+D_3(s) \tag{2b}$$

where

$$K_1 = \frac{1}{s} D_2(s)^{-1}D_1(s) \qquad \text{as} \qquad s\to\infty$$

= an m×m nonsingular matrix quotient

and

$$D_3(s) = \text{The matrix polynomial in } s \text{ with degree } (n-1)$$

$$= \sum_{j=1}^{n} D_{3,j} s^{n-j}$$

$$= \text{The left remainder of } D_1(s)$$

$D_2(s)$ and $D_3(s)$ have the same degree of $(n-1)$. If $D_3(s)$ is nonsingular, the inversion of $D_2(s)^{-1}D_3(s)$ in Eq. (2a) becomes

$$D_3(s)^{-1}D_2(s) = H_2+D_3(s)^{-1}D_4(s) \tag{2c}$$

or

$$D_3(s) = D_2(s)H_2^{-1}-D_4(s)H_2^{-1} \tag{2d}$$

where

$$H_2 = D_3(s)^{-1}D_2(s) \quad \text{as} \quad s \to \infty$$

$$= \text{an } m\times m \text{ nonsingular matrix quotient}$$

and

$$D_4(s) = \text{The matrix polynomial in } s \text{ with degree } (n-2)$$

$$\triangleq \sum_{j=1}^{n-1} D_{4,j} s^{n-j-1}$$

Substituting Eq. (2d) into Eq. (2b) gives

$$D_1(s) = D_2(s)(sK_1+H_2^{-1})-D_4(s)H_2^{-1} \tag{2e}$$

$$= D_2(s)Q_1(s)-D_4(s)H_2^{-1}$$

where

$$Q_1(s) = sK_1+H_2^{-1}$$

In the same fashion, we can use $D_2(s)$ with degree of $(n-1)$, and $D_4(s)$ with degree of $(n-2)$, to generate another matrix polynomial $D_6(s)$ as

$$D_2(s) = D_4(s)Q_3(s)-D_6(s)H_4^{-1} \tag{2f}$$

where

$$Q_3(s) = sK_3+H_4^{-1}$$

In general we have

$$D_i(s) = D_{i+2}(s)Q_{i+1}(s)-D_{i+4}(s)H_{i+2}^{-1}, \qquad i = 0,2,4,\ldots,2n-2 \tag{3}$$

$$D_{2n+2}(s) = 0_m$$

where

$$D_0(s) \triangleq D_1(s)$$

$$Q_{i+1}(s) = sK_{i+1}+H_{i+2}^{-1}, \qquad i = 0,2,4,\ldots,2n-2$$

0_m is an m×m null matrix. The matrix coefficients of the matrix polynomial $D_i(s)$ are defined as $D_{i,j}$, j = 1,2,... . The procedure to evaluate the matrix coefficients $D_{i,j}$ can be easily accomplished by using the newly developed block Routh array with block Routh algorithm that is different from the matrix Routh array[4] with matrix Routh algorithm developed for expanding matrix continued fractions.[5,6]

The block Routh array is listed as follows.

$K_1 = D_{21}^{-1}D_{11}$	$D_{11} = A_{n+1}$	$D_{12} = A_n$	$\cdots$	$D_{1,n-1} = A_3$	$D_{1,n} = A_2$	$D_{1,n+1} = A_1$
	$D_{21} = B_n$	$D_{22} = B_{n-1}$	$\cdots$	$D_{2,n-1} = B_2$	$D_{2,n} = B_1$	
$H_2 = D_{31}^{-1}D_{21}$						
$K_3 = D_{41}^{-1}D_{21}$	$D_{31} \triangleq D_{12} - D_{22}K_1$	$D_{32} \triangleq D_{13} - D_{23}K_1$	$\cdots$	$D_{3,n-1}$	$D_{3,n}$	
	$D_{41} \triangleq D_{22} - D_{32}H_2$	$D_{42} \triangleq D_{23} - D_{33}H_2$	$\cdots$	$D_{4,n-1}$		
$H_4 = D_{51}^{-1}D_{41}$						
$K_5 = D_{61}^{-1}D_{41}$	$D_{51} \triangleq D_{22} - D_{42}K_3$	$D_{52} \triangleq D_{23} - D_{43}K_3$	$\cdots$	$D_{5,n-1}$		
	$D_{61} \triangleq D_{42} - D_{52}H_4$	$D_{62} \triangleq D_{43} - D_{53}H_4$	$\cdots$			
	$\cdots$	$\cdot$	$\cdots$			
	$D_{2n-2,1}$	$D_{2n-2,2}$				
$H_{2n-2} = D_{2n-1,1}^{-1}D_{2n-2,1}$						
$K_{2n-1} = D_{2n,1}^{-1}D_{2n-2,1}$	$D_{2n-1,1}$	$D_{2n-1,2}$				
	$D_{2n,1} \triangleq D_{2n-2,2} - D_{2n-1,2}H_{2n-2}$					
$H_{2n} = D_{2n+1,1}^{-1}D_{2n,1}$						
	$D_{2n+1,1} \triangleq D_{2n-2,2}$					(4a)

The procedure to construct the block Routh array is described as follows.

Enter the matrix coefficients of $D_1(s)$ and of $D_2(s)$ respectively as row 1 and row 2, and evaluate in succession by using the block Routh algorithm (to be shown) to obtain row 3 from rows 1 and 2, and row 4 from rows 2 and 3. Then rows 2 and 4 are used to generate row 5, also the rows 4 and 5 are used to generate row 6. The above processes to generate row 4 from rows 2 and 3, and row 5 from rows 2 and 4 are repeated and continued to the last row (2n+1). The block Routh algorithms is

(i) $K_1 = D_{21}^{-1}D_{11}$, rank $D_{21} = m$

$$D_{3,j} = D_{1,j+1} - D_{2,j+1}K_1, \qquad j = 1,2,\ldots,n$$

$H_2 = D_{31}^{-1}D_{21}$, rank $D_{31} = m$

$$D_{4,j} = D_{2,j+1} - D_{3,j+1}H_2, \qquad j = 1,2,\ldots,n-1$$

(ii)
$$\left.\begin{aligned} K_{i+1} &= D_{i+2,1}^{-1}D_{i,1}, \quad \text{rank } D_{i+2,1} = m \\ D_{i+3,j} &= D_{i,j+1} - D_{i+2,j+1}K_{i+1} \\ H_{i+2} &= D_{i+3,1}^{-1}D_{i+2,1}, \quad \text{rank } D_{i+3,1} = m \\ D_{i+4,j} &= D_{i+2,j+1} - D_{i+3,j+1}H_{i+2} \end{aligned}\right\} \begin{aligned} j &= 1,2,3,\ldots; \\ i &= 2,4,6,\ldots,2n-2 \end{aligned} \qquad (4b)$$

$$D_{2n+2,1} = 0_m$$

From the block Routh array we can construct a sequence of matrix polynomials $D_i(s)$, $i = 0,1,2,\ldots$, and we shall show that the matrix sequence

$\{D_0(s), D_2(s), D_4(s), \ldots, D_{2n-2}(s)\}$ is a matrix Sturm series of $T(s)$.

The matrix series in Eq. (3) can be modified and expressed in the form of a matrix Sturm series developed by Bitmead and Anderson[3] as:

$$D_i^*(s) = D_{i+2}^*(s)Q_{i+1}^*(s) - D_{i+4}^*(s), \quad i = 0,2,4,\ldots,2n-2 \tag{5a}$$

$$D_{2n+2}^*(s) = 0_m$$

where

$$D_0^*(s) = D_0(s) = D_1(s)$$

$$D_2^*(s) = D_2(s)$$

$$D_{4k}^*(s) = D_{4k}(s)\Big[\prod_{j=1}^{k} H_{4j-2}\Big]^{-1}, \qquad k = 1,2,3,\ldots$$

$$D_{4k+2}^*(s) = D_{4k+2}(s)\Big[\prod_{j=1}^{k} H_{4j}\Big]^{-1}, \quad k = 1,2,3,\ldots \tag{5b}$$

and

$$Q_1^*(s) = Q_1(s)$$

$$Q_3^*(s) = H_2Q_3(s)$$

$$Q_{4k+1}^*(s) = \Big[\prod_{j=1}^{k} H_{4j}\Big]Q_{4k+1}(s)\Big[\prod_{j=1}^{k} H_{4j-2}\Big]^{-1}, \qquad k = 1,2,3,\ldots$$

$$Q_{4k-1}^*(s) = \Big[\prod_{j=1}^{k} H_{4j-2}\Big]Q_{4k-1}(s)\Big[\prod_{j=1}^{k-1} H_{4j}\Big]^{-1}, \qquad k = 2,3,\ldots$$

$$Q_{i+1}(s) = sK_{i+1} + H_{i+2}^{-1}, \qquad i = 0,2,4,\ldots,2n-2 \tag{5c}$$

For example, when $i = 4$ in Eq. (5a), we have

$$D_4^*(s) = D_6^*(s)Q_5^*(s)-D_8^*(s) \tag{6a}$$

From Eqs. (5b) and (5c) we have

$$D_4^*(s) = D_4(s)H_2^{-1}$$

$$D_6^*(s) = D_6(s)H_4^{-1}$$

$$D_8^*(s) = D_8(s)H_6^{-1}H_2^{-1} \tag{6b}$$

and

$$Q_5^*(s) = H_4Q_5(s)H_2^{-1} \tag{6c}$$

Substituting Eqs. (6b) and (6c) into Eq. (6a) and simplifying it yields

$$D_4(s)H_2^{-1} = [D_6(s)H_4^{-1}][H_4Q_5(s)H_2^{-1}]-D_8(s)H_6^{-1}H_2^{-1} \tag{6d}$$

or

$$D_4(s) = D_6(s)Q_5(s)-D_8(s)H_6^{-1} \tag{6e}$$

Eq. (6e) is one of the matrix equations in Eq. (3). When i = 6, we have

$$D_6^*(s) = D_8^*(s)Q_7^*(s)-D_{10}^*(s) \tag{6f}$$

Substituting the corresponding $D_i^*(s)$ and $Q_i^*(s)$ in Eq. (5) into Eq. (6f) we

have

$$D_6(s)H_4^{-1} = [D_8(s)H_6^{-1}H_2^{-1}][H_2H_6Q_7(s)H_4^{-1}]-D_{10}(s)H_8^{-1}H_4^{-1} \tag{6g}$$

or

$$D_6(s) = D_8(s)Q_7(s)-D_{10}(s)H_8^{-1} \tag{6h}$$

Again, Eq. (6h) is a matrix equation in Eq. (3).

From Eqs. (5) and (6) we observe that each matrix polynomial $D_i^*(s)$ in the sequence of real matrix polynomials $\{D_0^*(s),\ D_2^*(s),\ \ldots,\ D_{2n-2}^*(s)\}$ in Eq. (5) is different from the $D_i(s)$ in $\{D_0(s),D_2(s),\ldots,D_{2n-2}(s)\}$ in Eq. (3) by various weighting constant matrices shown in Eq. (5b). Furthermore, both $Q_i^*(s)$ and $Q_i(s)$ are matrix polynomials, and $D_{i+2}^*(s)$ and $D_{i+2}(s)$ are non-singular matrix polynomials with degree [det $D_{i+4}^*(s)$] = degree [det $D_{i+4}(s)$] < degree [det $D_{i+2}^*(s)$] = degree [det $D_{i+2}(s)$]. Using Bitmead and Anderson's theorem,[3] we can conclude that the sequence of real matrix polynomials $\{D_0(s),D_2(s),\ldots,D_{2n-2}(s)\}$ as well as the sequence $\{D_0^*(s),D_2^*(s),\ldots,D_{2n-2}^*(s)\}$ is a matrix Sturm sequence.

III. Distribution of Real Roots

It is well known that the scalar Sturm sequence is often used to determine the number of real roots of a scalar polynomial on the real axis in the complex plane. Bitmead and Anderson[3] have extended the scalar Sturm theorem to the matrix Sturm theorem. Since the matrix sequence $\{D_0(s),D_2(s),\ldots,$ $D_{2n-2}(s)\}$ in Eq. (3) is a matrix Sturm sequence, the matrix Sturm theorem can be applied to determine the number of real poles of a matrix transfer

function or a real roots of a matrix polynomial. A simple criterion using the block quotients K_i and H_i in Eq. (4) is developed to determine the realization of an impedance matrix of RC networks.[6] The matrix Cauchy index[3] of a symmetric real rational matrix T(s) with $T^{-1}(s)$ exists and $\det T(0) \neq 0$ can be written as

$$I_a^b T(s) = \Delta_a^b[D_0^*(s)^{-1}D_2^*(s)]+\Delta_a^b[D_2^*(s)^{-1}D_4^*(s)]+\ldots+\Delta_a^b[D_{2n-2}^*(s)^{-1}D_{2n}^*(s)]$$
$$= \Delta_a^b[D_0(s)^{-1}D_2(s)]+\Delta_a^b[D_2(s)^{-1}D_4(s)H_2^{-1}]+\Delta_a^b[H_2D_4(s)^{-1}D_6(s)H_4^{-1}]$$
$$+ \Delta_a^b[H_4D_6^{-1}(s)D_8(s)H_6^{-1}H_2^{-1}]+\ldots \tag{7a}$$

where

$$\Delta_a^b[D_i^*(s)^{-1}D_{i+2}^*(s)] = \frac{1}{2}[\sigma(T_i(b))-\sigma(T_i(a))]$$

= Half the total changes of signature of $T_i(s)$ over (a,b). (7b)

$\sigma(T)$ is defined as the signature[9] of a matrix T and $\sigma(T) = \dim V^+ - \dim V^-$, where $\dim V^+$ is the largest possible dimension of any subspace on which T is positive definite and $\dim V^-$ is the largest dimension of any subspace on which T is negative definite.

The I_a^b T(s) in Eq. (7) can be expressed in terms of K_i and H_i obtained in Eq. (4) as follows. Letting $a = -\infty$ and $b = \infty$ and substituting $D_i^*(s)$ in Eq. (5) into Eq. (7) we have

$$I_a^b T(s) = \sigma(K_1)^{-1}+\sigma(H_2K_3)^{-1}+\sigma(H_4K_5H_2^{-1})^{-1}+\sigma(H_2H_6K_7H_4^{-1})^{-1}+\ldots$$
$$= \sum_{i=1,3,5,}^{2n-1} \sigma(M_i^{-1}) \tag{8a}$$

where

$$M_1 = K_1$$

$$M_3 = H_2K_3$$

$$M_{4\ell+1} = [\prod_{j=1}^{\ell} H_{4j}]K_{4\ell+1}[\prod_{j=1}^{\ell} H_{4j-2}]^{-1}, \qquad \ell = 1,2,3,\ldots$$

$$M_{4\ell-1} = [\prod_{j=1}^{\ell} H_{4j-2}]K_{4\ell-1}[\prod_{j=1}^{\ell-1} H_{4j}]^{-1}, \qquad \ell = 2,3,\ldots \tag{8b}$$

when $a = -\infty$ and $b=0$, we have

$$I_a^b\; T(s) = \frac{1}{2}[\sigma(K_1^*)^{-1}+\sigma(K_1)^{-1}] + \frac{1}{2}[\sigma(H_2K_3^*)^{-1}+\sigma(H_2K_3)^{-1}]$$

$$+ \frac{1}{2}[\sigma(H_4K_5^*H_2^{-1})^{-1}+\sigma(H_4K_5H_2^{-1})^{-1}]+\ldots$$

$$= \frac{1}{2}\{\sum_{i=1,3,5,}^{2n-1} \sigma(M_i^{*-1}) + \sum_{i=1,3,5,}^{2n-1} \sigma(M_i^{-1})\} \tag{8c}$$

where

$$M_1^* = K_1^* = D_{2,n}^{-1}D_{1,n+1}$$

$$M_3^* = H_2K_3^* = H_2(D_{4,n-1}^{-1}D_{2,n})$$

$$M_{4\ell+1}^* = [\prod_{j=1}^{\ell} H_{4j}]K_{4\ell+1}^*[\prod_{j=1}^{\ell} H_{4j-2}]^{-1}, \qquad \ell = 1,2,3,\ldots$$

$$M_{4\ell-1}^* = [\prod_{j=1}^{\ell} H_{4j-2}]K_{4\ell-1}^*[\prod_{j=1}^{\ell-1} H_{4j}]^{-1}, \qquad \ell = 2,3,\ldots$$

$$K_{2i+1}^* = D_{2(i+1),(n-i)}^{-1}D_{2i,(n+1-i)}, \qquad i = 1,2,3,\ldots,n-1 \tag{8d}$$

M_i are shown in Eq. (8b). When $a=0$ and $b=\infty$, we have

$$I_a^b\, T(s) = \frac{1}{2}[\sigma(K_1)^{-1}-\sigma(K_1^*)^{-1}] + \frac{1}{2}[\sigma(H_2K_3)^{-1}-\sigma(H_2K_3^*)^{-1}]$$

$$+ \frac{1}{2}[\sigma(H_4K_5H_2^{-1})^{-1}-\sigma(H_4K_5^*H_2^{-1})^{-1}]+\ldots$$

$$= \frac{1}{2}[\sum_{i=1,3,5,}^{2n-1} \sigma(M_i^{-1}) - \sum_{i=1,3,5,}^{2n-1} \sigma(M_i^{*-1})] \quad (8e)$$

M_i and M_i^* are shown in Eqs. (8b) and (8d).

The Cauchy Indices $I_a^b\, T(s)$ in Eq. (8) indicate the number by real poles of $T(s)$ in (a,b). When $2r$ $m{\times}m$ block quotients K_i and K_i^* are obtained in the block Routh array (that is constructed using an nth degree $T(s)$ with $T^{-1}(s)$ exists and det $T(0) \neq 0$) and the Cauchy index in Eq. (8c) equals to rm (where $r{\le}n$), then $T(s)$ is asymptotically stable and the multivariable system is an aperiodic system.[7] Furthermore, if all $D_{i+2}(s)^{-1}D_i(s)$ are symmetric real rational matrices of an aperiodic system, then $T(s)$ is relizable as the impedance of an m-port RC network.[3] This implies that, when r matrices $Q_i^*(s)$ in Eq. (5c) are symmetric and the Cauchy index in *Eq.* (8c) *is equal to rm*, $T(s)$ can be synthesized using RC elements.

When the number of distinct real roots of det $D_1(s)$ is of interest, a matrix polynomial $D_2(s)(\triangleq d[D_1(s)]/ds)$ is constructed. By using $T(s)=D_1(s)^{-1}D_2(s)$ and the above procedures, the number of distinct real roots of det $D_1(s)$ in (a,b) can be determined.

IV A Block Canonical Form

When the total number (2r) of the block quotients K_i and H_i in Eq. (4) equals to 2n, where n is the degree of $D_1(s)$ in the real rational matrix transfer function $T(s)(= D_1(s)^{-1}D_2(s))$, the left matrix fraction decomposition $D_1(s)^{-1}D_2(s)$ in Eq. (1) consists of left coprime matrix polynomials,[8] which may contain non-symmetric matrix coefficients. The $T(s)$ can be for-

mulated into a block canonical form as follows.

From Eq. (3) we have

$$D_2(s)^{-1}D_1(s) = Q_1(s)-D_2(s)^{-1}D_4(s)H_2^{-1}$$

$$D_4(s)^{-1}D_2(s) = Q_3(s)-D_4(s)^{-1}D_6(s)H_4^{-1}$$

$$D_6(s)^{-1}D_4(s) = Q_5(s)-D_6(s)^{-1}D_8(s)H_6^{-1}$$

$$\cdots$$

$$D_{2n}(s)^{-1}D_{2n-2}(s) = Q_{2n-1}(s)$$

$$D_{2n+1}(s)^{-1}D_{2n}(s) = H_{2n} \tag{9}$$

Successively substituting Eq. (9) into T(s) yields

$$\begin{aligned}
T(s) &= D_1(s)^{-1}D_2(s) = [D_2(s)^{-1}D_1(s)]^{-1} \\
&= [Q_1(s)-D_2(s)^{-1}D_4(s)H_2^{-1}]^{-1} \\
&= [Q_1(s)-[D_4(s)^{-1}D_2(s)]^{-1}H_2^{-1}]^{-1} \\
&= [Q_1(s)-[Q_3(s)-[D_6(s)^{-1}D_4(s)]^{-1}H_4^{-1}]^{-1}H_2^{-1}]^{-1} \\
&= \cdots \\
&= [Q_1(s)-[Q_3(s)-[Q_5(s)-[\ldots-[Q_{2n-1}(s)]^{-1}H_{2n}^{-1}\ldots]^{-1}H_6^{-1}]^{-1}H_4^{-1}]^{-1}H_2^{-1}]^{-1}
\end{aligned} \tag{10}$$

where

$$Q_{i+1}(s) = sK_{i+1}+H_{i+2}^{-1}, \qquad i = 0,2,4,\ldots$$

Eq. (10) is a block canonical form of T(s). The corresponding block diagram

is shown in Fig. 1 and the block state equation obtained from the block diagram can be written as

$$\dot{z} = Gz + Er$$

$$y = Fz \tag{11}$$

where

$$G = \begin{bmatrix} -(K_{2n-1}H_{2n})^{-1} & (K_{2n-3}H_{2n-2})^{-1} & \cdot & 0_m & 0_m & 0_m & 0_m \\ K_{2n-1}^{-1} & -(K_{2n-3}H_{2n-2})^{-1} & \cdot & 0_m & 0_m & 0_m & 0_m \\ 0_m & K_{2n-3}^{-1} & \cdot & 0_m & 0_m & 0_m & 0_m \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0_m & 0_m & \cdot & -(K_7H_8)^{-1} & (K_7H_6)^{-1} & 0_m & 0_m \\ 0_m & 0_m & \cdot & K_7^{-1} & -(K_5H_6)^{-1} & (K_3H_4)^{-1} & 0_m \\ 0_m & 0_m & \cdot & 0_m & K_5^{-1} & -(K_3H_4)^{-1} & (K_1H_2)^{-1} \\ 0_m & 0_m & \cdot & 0_m & 0_m & K_3^{-1} & -(K_1H_2)^{-1} \end{bmatrix}$$

$$z^T = [z_1^T, z_2^T, \ldots, z_n^T]$$

$$E^T = [0_m, 0_m, \ldots, 0_m, I_m]$$

$$F = [0_m, 0_m, \ldots, 0_m, K_1^{-1}]$$

The T in Eq. (11) designates transpose, z_i are m×1 vectors, r is an m×1 input vector and y is an m×1 output vector. G is a block tridiagonal matrix with block elements constructed by using K_i and H_i.

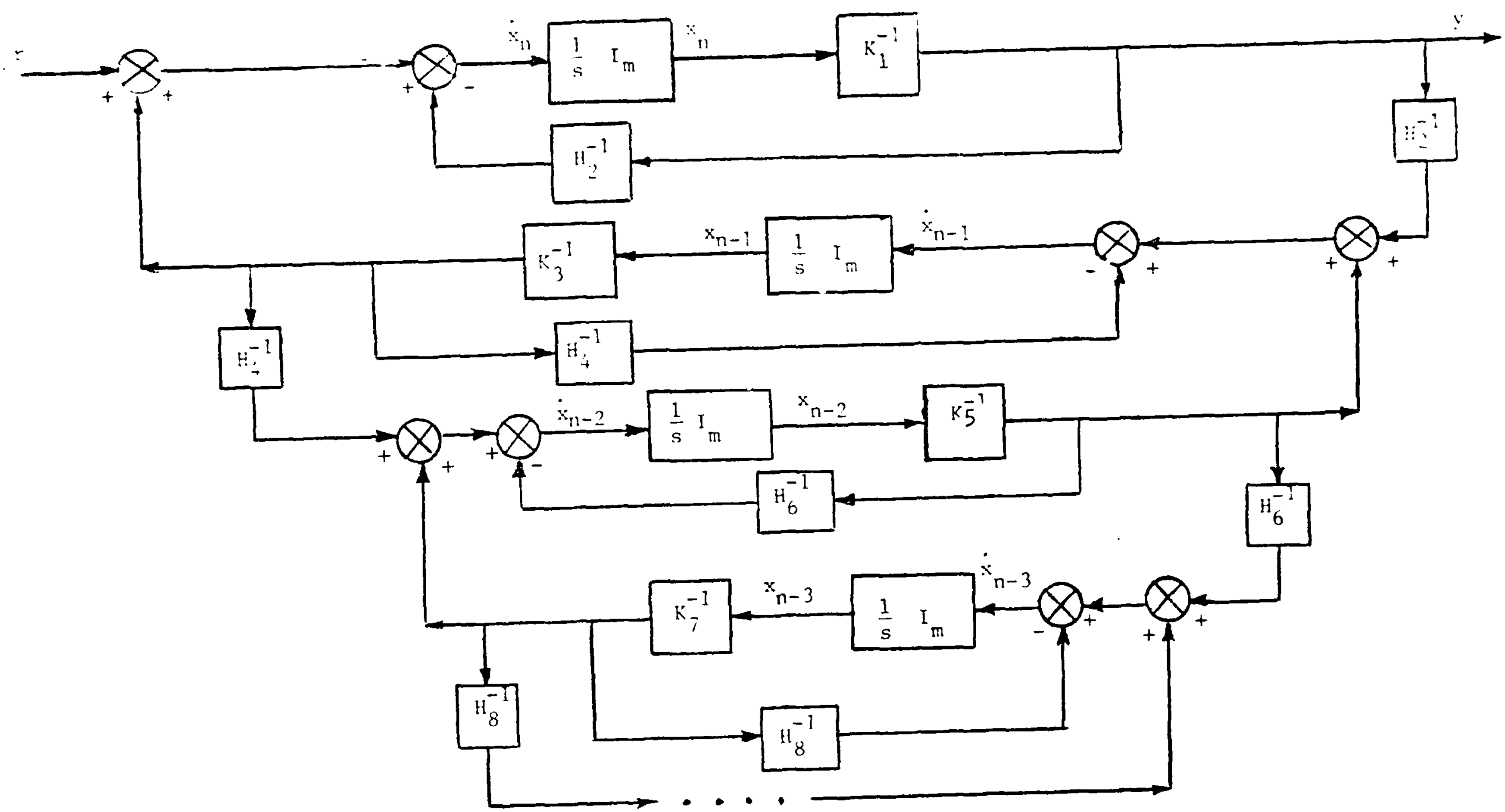


Fig. 1. Block diagram representation of a block canonical form.

An alternate block state equation that can be directly written from T(s) in Eq. (1) with $D_{11} = I_m$ is

$$\dot{x} = Ax + Br$$

$$y = Cx, \quad x(0) = 0_{nm\times 1} \tag{12}$$

where

$$A = \begin{bmatrix} 0_m & 0_m & . & 0_m & -D_{1,n+1} \\ I_m & 0_m & . & 0_m & -D_{1,n} \\ 0_m & I_m & . & 0_m & -D_{1,n-1} \\ . & . & . & . & . \\ 0_m & 0_m & . & I_m & -D_{1,2} \end{bmatrix}, \qquad B = \begin{bmatrix} D_{21} \\ D_{22} \\ D_{23} \\ . \\ D_{2,n} \end{bmatrix}, \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ x_n \end{bmatrix}$$

$$C = [0_m, \ 0_m, \ . \ 0_m, \ I_m]$$

$0_{nm\times 1}$ is an nm×1 null vector and I_m is an m×m identity matrix. x_i are m×1 vectors. The block state equation in Eq. (12) is an observable block Companion form. It is interesting to notice that the block linear transformation matrix between the block coordinates z in Eq. (11) and x in Eq. (12) can be directly written from the block Routh array in Eq. (4) as

$$x = Rz \tag{13}$$

$$
R = \begin{bmatrix}
D_{2n,1} & D_{2n-2,2} & \cdot & D_{8,n-3} & D_{6,n-2} & D_{4,n-1} & D_{2,n} \\
0_m & D_{2n-2,1} & \cdot & D_{8,n-4} & D_{6,n-3} & D_{4,n-2} & D_{2,n-1} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
0_m & 0_m & \cdot & D_{81} & D_{62} & D_{43} & D_{24} \\
0_m & 0_m & \cdot & 0_m & D_{61} & D_{42} & D_{23} \\
0_m & 0_m & \cdot & 0_m & 0_m & D_{41} & D_{22} \\
0_m & 0_m & \cdot & 0_m & 0_m & 0_m & D_{21}
\end{bmatrix}
$$

Substituting Eq. (13) into Eq. (11) and comparing the respective system matrix, input vector and output vector of Eqs. (11) and (12), we have

$$A = RGR^{-1} \tag{14a}$$

$$B = RE \tag{14b}$$

$$C = FR^{-1} \tag{14c}$$

When the total number (2r) of the block quotients K_i and H_i is equal to 2n, the block state equations in Eqs. (11) and (12) are the minimal realizations of T(s).

When 2r<2n, T(s) in Eq. (1) that does not consist of coprime matrix polynomials can be written as:

$$
\begin{aligned}
T(s) &= D_1(s)^{-1}D_2(s) = [D_{11}s^{n}+D_{12}s^{n-1}+\ldots+D_{1,n+1}]^{-1}[D_{21}s^{n-1}+D_{22}s^{n-2}+\ldots+D_{2,n}] \\
&= [C(s)P_1(s)]^{-1}[C(s)P_2(s)] \\
&= P_1(s)^{-1}P_2(s) = [P_{11}s^{r}+\ldots+P_{1,r+1}]^{-1}[P_{21}s^{r-1}+P_{22}s^{r-2}+\ldots+P_{2,r}]
\end{aligned} \tag{15}
$$

where

$$C(s) = C_{n+1-r}s^{n-r}+C_{n-r}s^{n-r-1}+\ldots+C_1 \quad \text{and}$$

$C(s)$ is a common matrix polynomial and $P_1(s)$ and $P_2(s)$ are left coprime. The $C(s)$ can be constructed from the matrix coefficients in the last non-vanishing row in the block Routh array. The matrix coefficients $P_{i,j}$ in Eq. (15) can be determined as follows: From Eq. (15) we have

$$D_1(s) = C(s)P_1(s) \tag{16a}$$

or

$$D_{11}s^n+D_{12}s^{n-1}+\ldots+D_{1,n+1} = [C_{n+1-r}s^{n-r}+C_{n-r}s^{n-r-1}+\ldots+C_1] \times$$
$$[P_{11}s^r+P_{12}s^{r-1}+\ldots+P_{1,r+1}]$$

Equating the matrix coefficients of the successive power of s we thus require

$$\begin{aligned} D_{11} &= C_{n+1-r}P_{11} \\ D_{12} &= C_{n+1-r}P_{12}+C_{n-r}P_{11} \\ D_{13} &= C_{n+1-r}P_{13}+C_{n-r}P_{12}+C_{n-r-1}P_{11} \\ &\ldots \end{aligned} \tag{16b}$$

If C_{n+1-r} is not singular, the P_{11} can be determined. Also, the $P_{1,j}$ can be obtained in succession by Eq. (16b). In the same fashion, $P_{2,j}$ can be

determined by comparing the matrix coefficients of the following matrix equation:

$$D_2(s) = C(s)P_2(s) \tag{16c}$$

or

$$D_{21}s^{n-1}+D_{22}s^{n-2}+\ldots+D_{2,n} = [C_{n+1-r}s^{n-r}+C_{n-r}s^{n-r-1}+\ldots+C_1] \times$$
$$[P_{21}s^{r-1}+P_{22}s^{r-2}+\ldots+P_{2,r}]$$

It is noticed that the block linear transformation matrix R in Eq. (13) can be constructed using the block elements in the new block Routh array that is generated from $T(s) = P_1(s)^{-1}P_2(s)$ with $P_{11} = I_m$ but not from those in the block Routh array generated from $T(s) = D_1(s)^{-1}D_2(s)$. It is also noted that, R is an upper block triangular matrix, the inversion of R can be obtained by an iterative method. For example, the inversion of an 3m×3m matrix R_3, which is obtained by partitioning the R in Eq. (13), is required. The product of R_3 and R_3^{-1} is written as

$$R_3R_3^{-1} = \begin{bmatrix} D_{61} & D_{42} & D_{23} \\ 0_m & D_{41} & D_{22} \\ 0_m & 0_m & D_{21} \end{bmatrix} \begin{bmatrix} D_{61}^{-1} & x & y \\ 0_m & D_{41}^{-1} & z \\ 0_m & 0_m & D_{21}^{-1} \end{bmatrix} = \begin{bmatrix} I_m & 0_m & 0_m \\ 0_m & I_m & 0_m \\ 0_m & 0_m & I_m \end{bmatrix} \tag{17a}$$

x,y and z are unknown block elements to be determined. Expanding Eq. (15) and solving the resulting matrix equations gives

$$D_{61}x+D_{42}D_{41}^{-1} = 0_m \rightarrow x = -D_{61}^{-1}D_{42}D_{41}^{-1} \tag{17b}$$

$$D_{41}z+D_{22}D_{21}^{-1} = 0_m \rightarrow z = -D_{41}^{-1}D_{22}D_{21}^{-1} \tag{17c}$$

$$D_{61}y+D_{42}z+D_{23}D_{21}^{-1} = 0_m \rightarrow y = -D_{61}^{-1}[D_{42}D_{41}^{-1}D_{22}D_{21}^{-1}+D_{23}D_{21}^{-1}] \tag{17d}$$

Thus, we can determine R_3^{-1}.

When $P_1(s)$ and $P_2(s)$ are relatively prime, the characteristic poles of this multivariable system T(s) are the zeros of det $P_1(s)$ but not det $D_1(s)$, and the transmission zeros of T(s) are the zeros of det $P_2(s)$ but not det $D_2(s)$.

V. An Illustrative Example

Given: A matrix transfer function of a multivariable system as

$$y(s) = T(s)R(s) \tag{18}$$

where

$$T(s) = D_1(s)^{-1}D_2(s) = [C(s)P_1(s)]^{-1}[C(s)P_2(s)]$$

$$= [D_{11}s^3+D_{12}s^2+D_{13}s+D_{14}]^{-1}[D_{21}s^2+D_{22}s+D_{23}]$$

$$= \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}s^3 + \begin{pmatrix} 166 & 71 \\ 70 & 37 \end{pmatrix}s^2 + \begin{pmatrix} 383 & 191 \\ 472 & 214 \end{pmatrix}s + \begin{pmatrix} 32 & 42 \\ 40 & 52 \end{pmatrix}\right]^{-1} \times$$

$$\left[\begin{pmatrix} 1 & -2 \\ -2 & 5 \end{pmatrix}s^2 + \begin{pmatrix} 19 & 35 \\ 8 & 16 \end{pmatrix}s + \begin{pmatrix} 54 & 52 \\ 68 & 64 \end{pmatrix}\right]$$

with $n = 3$ and $m = 2$.

Determine:

(i) The number of real poles of T(s).

(ii) The realizability of T(s) using an m-port RC network.

(iii) The common matrix polynomial C(s).

(iv) The minimal realization of T(s).

(v) A pair of left coprime matrix polynomials $P_1(s)$ and $P_2(s)$.

(vi) The characteristic poles and transmission zeros of this multivariable system.

To solve above problems we construct the block Routh array as

$$D_{11} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad D_{12} = \begin{pmatrix} 166 & 71 \\ 70 & 37 \end{pmatrix} \quad D_{13} = \begin{pmatrix} 383 & 191 \\ 472 & 214 \end{pmatrix} \quad D_{14} = \begin{pmatrix} 32 & 42 \\ 40 & 52 \end{pmatrix}$$

$$D_{21} = \begin{pmatrix} 1 & -2 \\ -2 & 5 \end{pmatrix} \quad D_{22} = \begin{pmatrix} 19 & 35 \\ 8 & 16 \end{pmatrix} \quad D_{23} = \begin{pmatrix} 54 & 52 \\ 68 & 64 \end{pmatrix}$$

$$D_{31} = \begin{pmatrix} 1 & -2 \\ -2 & 5 \end{pmatrix} \quad D_{32} = \begin{pmatrix} 9 & 31 \\ 4 & 14 \end{pmatrix} \quad D_{33} = \begin{pmatrix} 32 & 42 \\ 40 & 52 \end{pmatrix}$$

$$D_{41} = \begin{pmatrix} 10 & 4 \\ 4 & 2 \end{pmatrix} \quad D_{42} = \begin{pmatrix} 22 & 10 \\ 28 & 12 \end{pmatrix}$$

$$D_{51} = \begin{pmatrix} 24 & 22 \\ 10 & 10 \end{pmatrix} \quad D_{52} = \begin{pmatrix} 54 & 52 \\ 68 & 64 \end{pmatrix}$$

$$D_{61} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{19a}$$

$$K_1 = D_{21}^{-1}D_{11} = \begin{pmatrix} 5 & 2 \\ 2 & 1 \end{pmatrix}, \qquad H_2 = D_{31}^{-1}D_{21} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$K_3 = D_{41}^{-1}D_{21} = \begin{pmatrix} 2.5 & -6 \\ -6 & 14.5 \end{pmatrix}, \qquad H_4 = D_{51}^{-1}D_{41} = \begin{pmatrix} 0.6 & -0.2 \\ -0.2 & 0.4 \end{pmatrix}$$

Because $D_{61} = 0_2$, the block Routh array terminates prematurely, and we have $2r$ $(r=2<n=4)=4$ block quotients. From the array we can also determine

$$K_1^* = D_{23}^{-1}D_{14} = \begin{pmatrix} 0.4 & 0.2 \\ 0.2 & 0.6 \end{pmatrix}, \qquad K_3^* = D_{42}^{-1}D_{23} = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \tag{19b}$$

Substituting K_i, K_i^* and H_i into Eq. (8c) yields

$$I_{-\infty}^{0} = \frac{1}{2}[\sigma(M_1^{*-1})+\sigma(M_3^{*-1})+\sigma(M_1^{-1})+\sigma(M_3^{-1})] = 4 \tag{19c}$$

where

$$M_1^* = K_1^* = \begin{pmatrix} 0.4 & 0.2 \\ 0.2 & 0.6 \end{pmatrix}, \qquad M_3^* = H_2K_3^* = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$$

$$M_1 = K_1 = \begin{pmatrix} 5 & 2 \\ 2 & 1 \end{pmatrix}, \qquad M_3 = H_2K_3 = \begin{pmatrix} 2.5 & -6 \\ -6 & 14.5 \end{pmatrix}$$

From Eq. (19c) we conclude that $T(s)$ has four negative real poles in $(-\infty,0)$.

To determine the realizability of $T(s)$ using passive RC network, we have to test whether $T(s)$ in Eq. (18) is a symmetric matrix. This can be easily accomplished by checking the matrices $Q_1^*(s)$ and $Q_3^*(s)$ in Eq. (5c) using K_i and H_i in Eq. (19a) as follows:

$$Q_1^*(s) = sK_1+H_2^{-1} = s\begin{pmatrix} 5 & 2 \\ 2 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$Q_3^*(s) = H_2(sK_3+H_4^{-1}) = s\begin{pmatrix} 2.5 & -6 \\ -6 & 14.5 \end{pmatrix} + \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \tag{19d}$$

Both $Q_1^*(s)$ and $Q_3^*(s)$ are symmetric, therefore T(s) is symmetric. Because the number of the Cauchy index in Eq. (19c) equals to rm (=4) and T(s) is symmetric, T(s) can be synthesized using passive RC elements.

Since the block Routh array in Eq. (19a) terminates prematurely, we can write the C(s) in Eq. (18) as

$$C(s) = C_{51}s+C_{52} = \begin{pmatrix} 24 & 22 \\ 10 & 10 \end{pmatrix} s+ \begin{pmatrix} 54 & 52 \\ 68 & 64 \end{pmatrix} = \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} s+ \begin{pmatrix} 1 & 3 \\ 2 & 2 \end{pmatrix}\right]\begin{pmatrix} 24 & 22 \\ 10 & 10 \end{pmatrix} \tag{19e}$$

To determine the minimal realization of T(s), we construct the block state equation in Eq. (11) using the K_i and H_i in Eq. (19a) as

$$\dot{z} = Gz + Er \tag{19f}$$

$$y = Fz$$

where

$$G = \begin{bmatrix} -(K_3H_4)^{-1} & (K_1H_2)^{-1} \\ K_3^{-1} & -(K_1H_2)^{-1} \end{bmatrix} = \begin{bmatrix} -\begin{pmatrix} 140 & 58 \\ 130 & 54 \end{pmatrix} & \begin{pmatrix} 1 & -2 \\ -2 & 5 \end{pmatrix} \\ \begin{pmatrix} 58 & 24 \\ 24 & 10 \end{pmatrix} & -\begin{pmatrix} 1 & -2 \\ -2 & 5 \end{pmatrix} \end{bmatrix}$$

$$E = \begin{bmatrix} 0_2 \\ I_2 \end{bmatrix} = \begin{bmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \end{bmatrix}, \quad F = [0_2, K_1^{-1}] = \left[\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 1 & -2 \\ -2 & 5 \end{pmatrix}\right]$$

Eq. (19f) is the minimal realization of T(s).

The left coprime matrix polynomials $P_1(s)$ and $P_2(s)$ can be obtained from Eq. (16). The C(s) is modified to ensure that $P_{11} = I_2$ as follows

$$C(s) = I_2 s + C_{52}C_{51}^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} s + \begin{pmatrix} 1 & 3 \\ 2 & 2 \end{pmatrix} \tag{19g}$$

The required $P_1(s)$ and $P_2(s)$ are

$$P_1(s) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} s^2 + \begin{pmatrix} 165 & 68 \\ 68 & 35 \end{pmatrix} s + \begin{pmatrix} 14 & 18 \\ 6 & 8 \end{pmatrix} \tag{19h}$$

and

$$P_2(s) = \begin{pmatrix} 1 & -2 \\ -2 & 5 \end{pmatrix} s + \begin{pmatrix} 24 & 22 \\ 10 & 10 \end{pmatrix} \tag{19i}$$

Note that the matrix coefficients in $P_1(s)$ and $P_2(s)$ are not all symmetric but $T(s) = P_1(s)^{-1}P_2(s)$ is a symmetric matrix.

The characteristic poles of this multivariable system are the zeros of $\det P_1(s) = 0$, or

$$s_1 = -0.02739, \quad s_2 = -0.127864, \quad s_3 = -5.88774 \quad \text{and} \quad s_4 = -193.957.$$

The transmission zeros of this multivariable system are the zeros of $\det P_2(s) = 0$, or

$$s_1 = -0.10315 \quad \text{and} \quad s_2 = -193.89685 .$$

VI. Conclusion

An algebraic method has been developed for constructing a matrix Sturm series and for establishing a block canonical form of a matrix transfer function. A simple and effective block Routh algorithm has been developed to construct the block Routh array and the block quotients. The block quotients have been used to determine the number of real roots of a matrix polynomial, and to determine the realization of a driving-point RC impedance matrix. The minimal realizations of a matrix transfer function have been formulated to the block state equations in the block tridiagonal form and in the observable block companion form. As a result, a pair of left coprime matrix polynomials can be obtained, and the characteristic poles as well as the transmission zeros of a multivariable system can be determined.

References

[1] F. R. Gantmacher, The Theory of Matrices, Chelsea, New York, 1959.

[2] S. Barnett and D. D. Siljak, "Routh's Algorithm: A Centennial Survey", SIAM Review, Vol. 19, No. 3, July 1977, pp. 472-489.

[3] R. R. Bitmead and B. D. O. Anderson, "The Matrix Cauchy Index: Properties and Applications", SIAM Appl. Math., Vol. 33, No. 4, December 1977, pp. 655-672.

[4] L. S. Shieh, C. D. Shih and R. E. Yates, "Some Sufficient and Some Necessary Conditions for the Stability of Multivariable Systems", ASME Trans. J. of Dynamic Systems, Measurement and Control, Vol. 100, September 1978, pp. 214-218.

[5] L. S. Shieh and F. F. Gaudiano, "Some Properties and Applications of Matrix Continued Fractions", IEEE Trans. Circuits and Systems, Vol. CAS-22, September 1975, pp. 721-728.

[6] R. W. Newcomb, Linear Multiport Synthesis, McGraw-Hill, New York, 1966.

[7] A. T. Fuller, "Conditions for Aperiodicity in Linear Systems", British J. Appl. Phys., Vol. 6, 1955, pp. 195-198, pp. 450-451.

[8] S. Barnett, Matrices in Control Theory, Van Nostrand Reinhold Co., London, 1971.

[9] K. Hoffman and R. Kunze, Linear Algebra, Prentice-Hall, N. J. 1972, pp. 370-372.

CHAPTER V

CONCLUSION

The dominant-data matching method for analog pitch control system design has been successfully extended to design digital controller for the semi-active terminal homing missile system. Various digital filters have been designed and successfully tested in the 6 Degree-of-Freedom Terminal Homing Simulation Program at the MIRADCOM Laboratories.

The developed direct-decoupling method has been successfully applied to design an analog multivariable gas turbine system and a multivariable paper making machine. It is believed that this method can be further extended to digital redesign of the coupled row and yaw control system of the semi-active terminal homing missile system.

The properties of the newly developed Sturm series and block canonical form have been discussed. It is believed that the block canonical form can be further extended to synthesizing a multi-port network function without using integrators.

Other new findings of this research are reported in the appendix.

Comput & Elect. Engng Vol. 6, pp. 3–17
© Pergamon Press Ltd., 1979 Printed in Great Britain

0045-7906/79/0301-0003/$02.00/0

SOLVING INVERSE LAPLACE TRANSFORM, LINEAR AND NONLINEAR STATE EQUATIONS USING BLOCK-PULSE FUNCTIONS

L. S. SHIEH
Department of Electrical Engineering, University of Houston, Houston, TX 77004, U.S.A.

R. E. YATES
Guidance and Control Directorate, U.S. Army Missile Research and Development Command, Redstone Arsenal, AL 35809, U.S.A.

and

J. M. NAVARRO
Departamento de Ingenieria Electronica, Instituto Universitario Politecnico, Barquisimeto, Venezuela

(*Received* 28 *June* 1978; *received for publication* 12 *October* 1978)

Abstract—A recursive algorithm is developed for solving the inverse Laplace transform, linear and nonlinear state equations using block-pulse functions. The relationships between the solution of the continuous-time state equation using block-pulse functions and that of the equivalent discrete-time state equation using trapezoidal rule are investigated. A complete computer program is presented for solving the differential equations of linear and nonlinear state equations using block-pulse functions.

1. INTRODUCTION

An accurate description of a practical system (for example, a semiactive terminal homing missile system[1]) often results in a high order transfer function with very large coefficients and/or a high order linear and nonlinear time-invariant and/or time-varying state equation for which the commonly used numerical integration methods (e.g. the Runge–Kutta method[2]) may fail to determine the time response. Recently, an alternate method[3] that uses the block-pulse functions has been developed for solving the linear time-invariant state equations. In this paper, the method due to Shieh[3] and others is reviewed and extended to solve the inverse Laplace transform, linear and nonlinear state equations. Also, the relationships between the solution of the continuous-time state equations using block-pulse functions and that of the equivalent discrete-time state equations using the trapezoidal rule[3, 4] are further investigated. A complete computer program based on the proposed method is presented to solve the inverse Laplace transform, linear and nonlinear state equations using block-pulse functions. Several illustrative numerical examples are included to demonstrate the superiority of the new method.

2. MAIN RESULTS

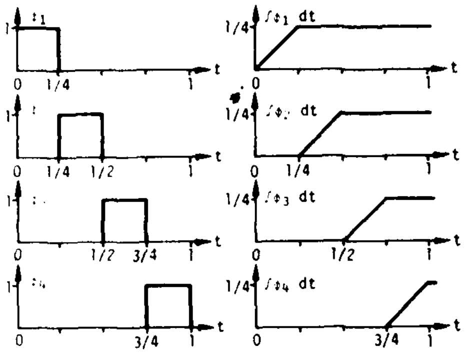
Consider a linear time-invariant state equation

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1a}$$

and

$$x(0) = x_0 \tag{1b}$$

where A is an $n \times n$ system matrix, B is an $n \times r$ input matrix, $x(t)$ is a state vector of n components, $u(t)$ is a vector of r input functions, and $x(0)$ is the initial state vector. The piecewise-constant solution of (1) can be obtained by using the block-pulse functions $\phi_j(t)$ for $j = 1, 2, \ldots, m$. Each block-pulse function $\phi_j(t)$ is defined by $\phi_j(t) = 1$ for $(j-1)T \le t < jT$, and $\phi_j(t) = 0$ for other cases. The term T is a time increment or a sampling period, and m is the number of the discrete-time solutions of interest. The block-pulse functions $\phi_j(t)$ for $j = 1, 2, 3, 4$ are shown in Fig. 1.



$$\begin{bmatrix} \int\phi_1\,dt \\ \int\phi_2\,dt \\ \int\phi_3\,dt \\ \int\phi_4\,dt \end{bmatrix} = \frac{1}{4}\begin{bmatrix} \frac{1}{2} & 1 & 1 & 1 \\ 0 & \frac{1}{2} & 1 & 1 \\ 0 & 0 & \frac{1}{2} & 1 \\ 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{bmatrix}$$

Fig. 1. The block-pulse functions and their integrations

The piecewise-constant solution and the discrete-time solution of (1) in the interval $(j-1)T < t < jT$ are defined as the column vector C_j and the $x^*(jT)$, respectively. The C_j and $x^*(jT)$ can be determined from the recursive algorithms shown in the following steps:

Step 1. Approximate the input vector $u(t)$ that has r input functions using the trapezoidal rule. The columns vectors L_j in the $r \times m$ matrix $L = [L_1, L_2, \ldots, L_m]$ that is the approximate input functions are

$$L_j = \frac{1}{2}[u(jT) + u((j-1)T)] \quad \text{for} \quad j = 1, 2, \ldots, m \tag{2}$$

$$= \text{average value of } u(t) \text{ over the interval } (j-1)T < t \le jT.$$

Step 2. Evaluate an $n \times m$ matrix $K = [K_1, K_2, \ldots, K_m]$. The column vectors K_j are

$$K_j = Ax(0) + BL_j \quad \text{for} \quad j = 1, 2, \ldots, m. \tag{3}$$

Step 3. Determine an $n \times m$ matrix $D = [D_1, D_2, \ldots, D_m]$. The column vectors D_j are:

$$D_1 = \frac{1}{T} R_1 K_1 \tag{4a}$$

$$D_j = (I_n + R_2)D_{j-1} + \frac{1}{T} R_1(K_j - K_{j-1}) \quad \text{for} \quad j = 2, 3, \ldots, m \tag{4b}$$

where

$$R_1 = \left(\frac{1}{T} I_n - \frac{1}{2} A\right)^{-1} = T\left(I_n - \frac{1}{2} AT\right)^{-1} \tag{4c}$$

$$R_2 = R_1 A$$

$$I_n = \text{an } n \times n \text{ identity matrix.}$$

Step 4. Obtain the $n \times m$ required piecewise-constant solution matrix

$C = [C_1, C_2, \ldots, C_m]$. The column vectors C_j are

$$C_1 = \frac{T}{2} D_1 + x(0) \tag{5a}$$

$$C_j = C_{j-1} + \frac{T}{2}(D_{j-1} + D_j) \quad \text{for} \quad j = 2, 3, \ldots, m\,. \tag{5b}$$

The required piecewise-constant solution of (1) is

$$x(t) = C\phi \tag{5c}$$

where $\phi = [\phi_1, \phi_2, \ldots, \phi_m]'$. The prime designates the transpose, and the ϕ_i is a block-pulse function.

Step 5. Determine the required approximate discrete-time solution $x^*(t)$ using the reversed process of the trapezoidal rule. The required solution $x^*(t)$ at discrete-time $t = (j+1)T$ is

$$x^*((j+1)T) = -x^*(jT) + 2C((j+1)T) \quad \text{for} \quad j = 0, 1, 2, \ldots, m-1$$

where

$$x^*(0) = x(0) \quad \text{and} \quad C((j+1)T) = C_{j+1}. \tag{6}$$

The expression $x^*(jT)$ is the approximate discrete-time solution of the $x(t)$ in (1). The accuracy of the approximation depends heavily upon the chosen sampling period T. A complete computer program, based on the algorithms in (2)–(6) is presented in this paper to obtain the solution $x^*(jT)$ in (6).

Because the trapezoidal rule is applied to approximate the input function in (2), it is interesting to investigate the relationships between the solution of the continuous-time state equation using block-pulse functions and that of an equivalent discrete-time state equation using the trapezoidal rule [4, 5].

When $u(t) = 0$ in (1), Shieh *et al.* [3] have shown that the equivalent discrete-time state equation of the continuous-time state equation in (1) is

$$x^*((j+1)T) = Gx^*(jT) \tag{7a}$$

$$x^*(0) = x_0$$

where

$$G = I_n + R_2 = I_n + \left(I_n - \frac{1}{2}AT\right)^{-1} AT \tag{7b}$$

$$= \left(I_n + \frac{1}{2}AT\right)\left(I_n - \frac{1}{2}AT\right)^{-1} = \left(I_n - \frac{1}{2}AT\right)^{-1}\left(I_n + \frac{1}{2}AT\right). \tag{7c}$$

The discrete-time solution of (7) is

$$x^*(jT) = G^j x(0) \quad \text{for} \quad j = 0, 1, 2 \ldots \tag{8}$$

In this paper, the derivation is extended to a more general case (Shieh *et al.* [3]), that is, $u(t) \neq 0$. To simplify the expression, the T in the following discrete-time state equations and difference equations is dropped. Also, the vectors L_j are expressed by $L(j)$, K_j by $K(j)$, D_j by $D(j)$ and C_j by $C(j)$.

When $u(t) \neq 0$, (2) to (5) can be expressed by a set of difference equations

$$D(1) = \frac{1}{T} R_1 K(1) = \frac{1}{T} R_1 A x(0) + \frac{1}{T} R_1 B L(1) \tag{9a}$$

$$D(j+1) = (I_n + R_2) D(j) + \frac{1}{T} R_1 B [L(j+1) - L(j)] \quad \text{for} \quad j = 1, 2, \ldots, m \tag{9b}$$

and

$$C(1) = \frac{1}{2}(2I_n + R_2)x(0) + \frac{1}{2} R_1 B L(1) \tag{10a}$$

$$C(j+1) = C(j) + \frac{T}{2}[D(j+1) + D(j)] \quad \text{for} \quad j = 1, 2, \ldots m. \tag{10b}$$

Substituting (9) into (10) and using (6) will yield the required discrete-time solution $x^*(j)$, or

$$x^*(j) = \phi^*(j)x(0) + \sum_{i=1}^{j} \phi^*(j-i)R_1BL(i)$$

$$= \phi^*(j)x(0) + \sum_{i=0}^{j-1} \phi^*(j-i-1)Hu^*(i) \quad \text{for} \quad j = 1, 2, \ldots \tag{11a}$$

where $\phi^*(j)$ = The transition matrix of the discrete-time system

$$= \left[\left(I_n - \frac{1}{2}AT\right)^{-1}\left(I_n + \frac{1}{2}AT\right)\right]^j$$

$$R_1 = T\left(I_n - \frac{1}{2}AT\right)^{-1}$$

$$H = R_1B$$

$$u^*(i) = \frac{1}{2}[u(i+1) + u(i)] \tag{11b}$$

$$T = \text{sampling period.}$$

From (11a) the approximate discrete-time state equation can be written for the continuous-time state equation in (1) as

$$x^*(j+1) = Gx^*(j) + Hu^*(j) \tag{12a}$$

$$x^*(0) = x(0) \tag{12b}$$

where

$$G = \left(I_n - \frac{1}{2}AT\right)^{-1}\left(I_n + \frac{1}{2}AT\right)$$

$$H = T\left(I_n - \frac{1}{2}AT\right)^{-1}B$$

$$u^*(j) = \frac{1}{2}[u(j+1) + u(j)]. \tag{12c}$$

It is believed that the modeling of the discrete-time state equation in (12) from the continuous-time state equation using block-pulse function is new. If the Z transformation is performed on both (1) and (12), we have the respective functions as

$$zX(z) - zX(0) = AX(z) + BU(z) \tag{13}$$

and

$$zX^*(z) - zX^*(0) = GX^*(z) + HU^*(z) \tag{14}$$

where

$$U^*(z) = \frac{1}{2}[zU(z) + U(z)].$$

Substituting the G and H in (12) into (14) and simplifying yields

$$\frac{2}{T}\frac{(z-1)}{(z+1)}X^*(z) - \frac{2}{T}\frac{z}{(z+1)}\left(I_n - \frac{1}{2}AT\right)X^*(0) = AX^*(z) + BU(z) \tag{15}$$

where $X^*(0) = X(0)$.

Comparing (13) and (15) and assuming that $X^*(z)$ is the approximate function of $X(z)$, we have

$$Z[\dot{x}(t)] \simeq \frac{2}{T}\frac{(z-1)}{(z+1)}X^*(z) - \frac{2}{T}\frac{z}{(z+1)}\left(I_n - \frac{1}{2}AT\right)X(0). \tag{16}$$

Equation (16) is the approximate numerical differentiator that is often used to determine the inverse Laplace transform of a continuous-time state equation[4, 5]. Thus, the solution of a linear time invariant state equation can be obtained from the recursive algorithms in (2)–(6), or from the matrix equations in (11). For linear and nonlinear time-varying systems, the frozen-time and frozen-state approach[6] is applied to solve the linear and nonlinear problems using block-pulse functions. In other words, when an independent variable t and the time dependent state variables $x_i(t)$ appear in the system matrix in (1) at stage j, the time variable t is considered as a frozen time by letting $t = jT$. The state variables considered as frozen states $x_i(jT)$ in the time intervals $jT \le t \le (j+1)T$. Substituting these constants $t = jT$ and $x_i(t) = x_i(jT)$ into the system matrix in (1) and using $x(jT)$ as the initial vector yields the time-invariant state equation in the time interval $jT \le t \le (j+1)T$. Thus, the proposed method can be applied to evaluate the solution $x(t)$ at $t = (j+1)T$. Repeating the processes we have the required discrete-time solutions for the linear and nonlinear state equations. A complete computer program based upon the above approach is presented in this paper for solving the inverse Laplace transform, linear and nonlinear state equations.

3. FORMULATIONS OF STATE EQUATIONS AND ILLUSTRATIVE EXAMPLES

Consider that the impulse response of the following rational function is required:

$$\frac{Y(s)}{U(s)} = \frac{b_1 s^{n-1} + b_2 s^{n-2} + \cdots + b_n}{s^n + a_1 s^{n-1} + a_2 s^{n-2} + \cdots + a_n}. \tag{17}$$

Since the input function $U(s) = 1$, the required impulse response is the inverse Laplace transform of $Y(s)$. Also, the impulse function is a delta function. It cannot be realized because of its infinite amplitude at $t = 0$. Therefore, it is convenient to convert (17) into a zero-input state equation with initial conditions as

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{18a}$$

$$y(t) = Cx(t) \tag{18b}$$

$$x(0) = x_0 \tag{18c}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdot & 0 \\ 0 & 0 & 1 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ -a_n & -a_{n-1} & \cdot & \cdot & -a_1 \end{bmatrix}, \quad x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \cdot \\ x_n(t) \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ 0 \end{bmatrix}$$

$$C = I_n.$$

The output matrix C is chosen as an identity matrix so that the output functions are equal to the state variables. The initial vector $x(0)$ can be evaluated from the following matrix equation [2, 7]:

$$x(0) = D^{-1}b \tag{19a}$$

or

$$\begin{bmatrix} x(0) \\ x'(0) \\ x''(0) \\ \cdot \\ x^{(n-1)}(0) \end{bmatrix} = \begin{bmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \\ \cdot \\ x_n(0) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdot & 0 \\ a_1 & 1 & 0 & \cdot & 0 \\ a_2 & a_1 & 1 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n-1} & a_{n-2} & a_{n-3} & \cdot & 1 \end{bmatrix}^{-1} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdot \\ b_n \end{bmatrix} \tag{19b}$$

b is a vector constructed from the coefficients of the numerator polynomial in (17). A recursive algorithm[2] has been proposed to determine the initial conditions without finding the inversion of the square matrix D. An alternate method is proposed in this paper to determine the D^{-1} and the required initial vector.

To determine the D^{-1} indirectly, we construct the following matrix equation:

$$z = K(-a) \tag{20a}$$

or

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ \cdot \\ z_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdot & 0 & 0 \\ z_1 & 1 & 0 & 0 & \cdot & 0 & 0 \\ z_2 & z_1 & 1 & 0 & \cdot & 0 & 0 \\ z_3 & z_2 & z_1 & 1 & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ z_{n-1} & z_{n-2} & z_{n-3} & z_{n-4} & \cdot & z_1 & 1 \end{bmatrix} \begin{bmatrix} -a_1 \\ -a_2 \\ -a_3 \\ -a_4 \\ \cdot \\ -a_n \end{bmatrix}. \tag{20b}$$

The vector $(-a)$ is constructed from the coefficients of the denominator polynomial in (17) with negative signs. The matrix K is a lower triangular matrix with each diagonal entry assigned as unity, and other entries z_i are determined from the vector z. In other words, from the multiplication of the first row vector in K and the vector $(-a)$ we have the numerical value z_1. Then, we immediately substitute the z_1 into the lower diagonal entries and solve for z_2, and so on. The general algorithm is

$$z_0 = 1$$

$$z_j = -\sum_{i=1}^{j} z_{i-1} a_{j-i+1} \quad \text{for} \quad j = 1, 2, \ldots, n. \tag{20c}$$

The matrix K is the inversion of the matrix D in (19). Thus the required initial vector $x(0)$ can be determined in (19). If the input vector $u(t)$ in (17) can be expressed by analytical functions or a set of finite values at sampling points, (17) can be converted to a zero initial-state time-invariant state equation as

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{21a}$$

$$y(t) = Cx(t) \tag{21b}$$

$$x(0) = 0 \tag{21c}$$

where A is shown in (18); $B = [0, 0, \ldots, 0, 1]'$; $C = [b_n, b_{n-1}, \ldots, b_2, b_1]$; and $x(0) = [0, 0, \ldots, 0, 0]'$.

A practical system is the transfer function of the pitch control system of a semi-active terminal homing missile system[1] which is shown as an illustrative example as follows:

$$\frac{Y(s)}{U(s)} = \frac{b_0 s^{10} + b_1 s^9 + \cdots + b_9 s + b_{10}}{s^{11} + a_1 s^{10} + a_2 s^9 + \cdots + a_{10} s + a_{11}} \tag{22a}$$

where

$$
\begin{array}{ll}
a_1 = 1.9235540 \times 10^3 & b_0 = 0 \\
a_2 = 9.3162391 \times 10^5 & b_1 = 0 \\
a_3 = 2.9769507 \times 10^8 & b_2 = 0 \\
a_4 = 6.2316753 \times 10^{10} & b_3 = 0 \\
a_5 = 9.3603299 \times 10^{12} & b_4 = 0 \\
a_6 = 9.7499233 \times 10^{14} & b_5 = 1.4945233 \times 10^{11} \\
a_7 = 6.6673970 \times 10^{16} & b_6 = 2.5633964 \times 10^{14} \\
a_8 = 2.4204054 \times 10^{18} & b_7 = 5.0172120 \times 10^{16} \\
a_9 = 2.9119206 \times 10^{18} & b_8 = 2.9263443 \times 10^{18} \\
a_{10} = 2.4190474 \times 10^{19} & b_9 = 4.6100047 \times 10^{19} \\
a_{11} = 8.8021585 \times 10^{18} & b_{10} = 8.8021585 \times 10^{18}.
\end{array}
$$

It is desired to find the step response $U(s) = (1/s)$.

Equation (22a) can be formulated either in the form on (18) or that of (21). Attempts to solve this problem by the Runge–Kutta method[2] were unsuccessful even though the time increment was chosen as small as 10^{-4} sec. This is because the practical system consists of large coefficients in the transfer function. This normally results from large poles, for example, in (22a) there exists a small a_1 (which is the sum of all poles) and a large a_{11} (which is the product of all poles). This difficulty is overcome by the proposed method. Using the proposed computer program with time increment $DT = 0.2$ sec yields the unit step response curve shown in Fig. 2. For comparing the results of the proposed method and the Runge–Kutta method we apply both methods to the reduced third-order model of the original 11th-order system in (21a) (using the method of Shieh and Chen[8, 9]) to evaluate the unit-step responses:

$$\frac{Y_3^*(s)}{U(s)} = \frac{0.6920s^2 + 19.4692s + 3.7376}{s^3 + 0.9488s^2 + 10.1661s + 3.7376}. \tag{22b}$$

The response curves are shown in Fig. 2. From these results, we observe that the proposed method is superior to the Runge–Kutta method if the system consists of extremely large or small coefficients or the response curve of the system has many stiff slopes.

When a linear or nonlinear time-varying equation is given and the numerical solution is required, the given equation can be converted into a state equation in (1) with time-varying and

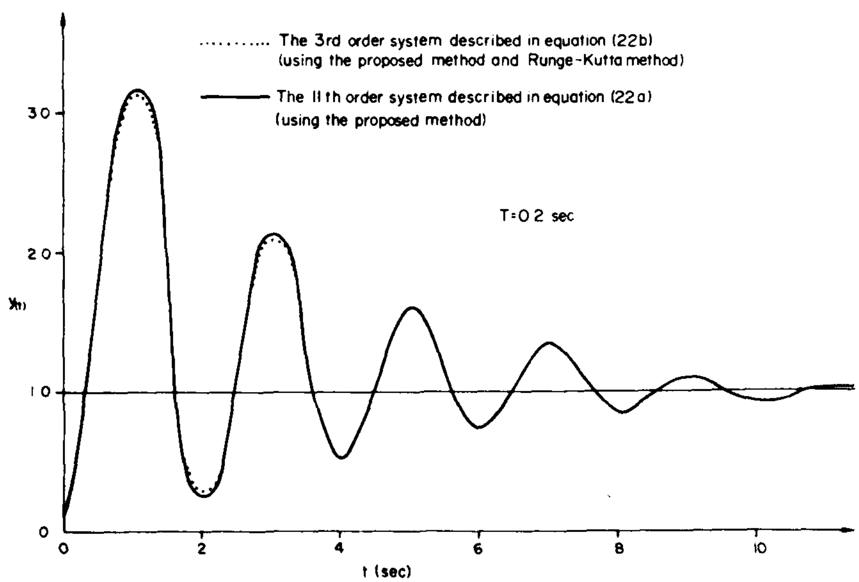


Fig. 2. Time responses of the systems described in eqns (22a) and (22b).

AD-A089 169 HOUSTON UNIV TX DEPT OF ELECTRICAL ENGINEERING F/G 16/4

ANALYTICAL DESIGN OF TERMINALLY GUIDED MISSILES.(U)

JAN 80 L SHIEH DAAK40-79-C-0061

UNCLASSIFIED NL

2 OF 2

AD A089169

END

DATE FILMED

10-80

DTIC

nonlinear entries in the system matrix. The proposed method, along with the frozen-time and frozen-state method, can be applied to determine the solution. To illustrate the procedure we use the following examples:

Given a nonlinear equation

$$\frac{d^2z(t)}{dt^2} - f(t)[1 - z(t)^2]\frac{dz(t)}{dt} + Kz(t) = Qu(t) \tag{23a}$$

$$z(0) = \alpha_1 \text{ and } \dot{z}(0) = \alpha_2$$

where $f(t)$ is a time-varying function, and K, Q and α_i are constants. Equation (23a) can be converted into a state equation by defining the state variables $x_1(t)$ and $x_2(t)$ as

$$x_1(t) = z(t)$$

$$x_2(t) = \dot{z}(t). \tag{23b}$$

The corresponding state equation is

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -K & f(t)[1 - x_1^2(t)] \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ Q \end{bmatrix} u(t) \tag{23c}$$

$$\begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}. \tag{23d}$$

When $f(t) = 1$, $K = 1$ and $Q = 0$, the nonlinear equation is the Van der Pol equation[10]. If the output functions $y_1(t)$ and $y_2(t)$ are assigned as $x_1(t)$ and $x_2(t)$, then the state equation is

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 1 - x_1^2(t) \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t) \tag{24a}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \tag{24b}$$

$$\begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}. \tag{24c}$$

The proposed method that uses the block-pulse functions and the frozen-time and frozen-state approach can be used to solve (24) for determining the trajectories $x_1(t)$ and $x_2(t)$. In other words, substituting $x(0)$ in (24c) into the system matrix in (24a) results in a time-invariant state equation in the form of (1). Thus, the developed recursive algorithm in (2)–(6) can be applied to evaluate $x^*(T)$ that is the required discrete-time solution $x(t)$ at $t = T$. Then, using the $x^*(T)$ obtained as the new initial vector in (24c) and again substituting the $x^*(T)$ into the system matrix in (24a) to obtain the new time-invariant system matrix for evaluating the new solution $x^*(T)$ that is the required solution $x(t)$ at $t = 2T$, and so on. The trajectories of the nonlinear equation as a result of different sets of initial conditions are shown in Fig. 3.

4. CONCLUSION

The recursive algorithm for solving linear time-invariant state equations using block-pulse functions has been extended for solving the inverse Laplace transform, linear and nonlinear time-invariant and time-varying state equations. The relationships between the solution of the continuous-time state equations using block-pulse functions and that of the equivalent discrete-time state equations using trapezoidal rule have been investigated. It is shown that the discrete-time solutions of both methods are identical. An approximate numerical differentiator has also been derived. A complete computer program, based on the derived recursive algorithm using block-pulse functions and the frozen-time and frozen-state approach, has been written for solving the inverse Laplace transform, linear and nonlinear state equations. It has been shown that the proposed method is superior to the Runge–Kutta method if the system consists of stiff functions.

Acknowledgement—This work was supported in part by U.S. Army Missile Research and Development Command, DAAK 40-78-C-0017.

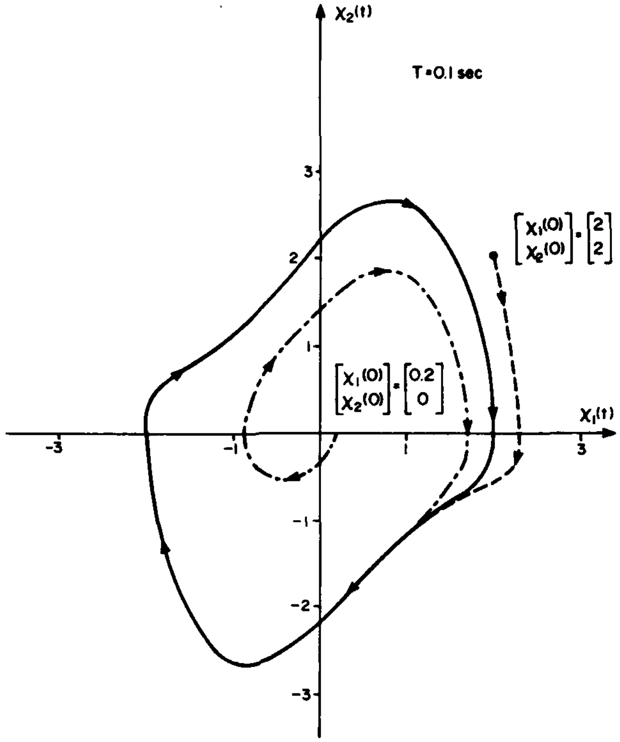


Fig. 3. Phase-plane diagram of the system described in eqn (24).

REFERENCES

1. J. T. Bosley, Digital realization of the T-6 missile analog autopilot. *Final Rep.*, U.S. Army Missile Command, DAAK40-77-C-0048, TGT-001, May 1977.
2. L. S. Shieh and C. F. Chen, Accurate determination of the frequency-to-time-domain matrix and its application to the inverse Laplace transform of high-order systems. *Comput. Elect. Engng* 4, 161–166 (1977).
3. L. S. Shieh, C. K. Yeung and B. C. McInnis, Solution of state-space equations via block-pulse functions. *Int. J. Control* 28(3), 383–392 (1978).
4. B. C. Kuo, *Analysis and Synthesis of Sampled-Data Control Systems*. Prentice-Hall, New Jerrsey (1963).
5. B. C. Kuo, *Digital Control Systems*, pp. 256–259. SRL, Champaign, Illinois (1977).
6. R. Boxer and S. Thaler, A simplified method of solving linear and nonlinear systems. *Proc. IRE* 44(1), 89–101 (1956).
7. C. F. Chen and R. E. Yates, A new matrix formula for the inverse Laplace transformation. *ASME Trans., J. Basic Engng* 269–272 (June 1967).
8. L. S. Shieh, An algebraic approach to system identification and compensator design. Ph.D. Dissertation, University of Houston, Houston, Texas, December 1970.
9. C. F. Chen and L. S. Shieh, A novel approach to linear model simplification. *Int. J. Control* 8(6), 561–570 (1968).
10. D. Graphan and D. McRuer, *Analysis of Nonlinear Control Systems*, pp. 46–76. Wiley, New York (1961).

APPENDIX

This program is used to solve the inverse Laplace transform, linear and nonlinear, time-invariant and time-varying state equations. The details to prepare the input cards can be illustrated by the following examples.

Example 1. For the following transfer function:

$$y(s) = \frac{s^2 + 8s + 12}{s^3 + 3s^2 + 10s + 0}. \tag{A1}$$

The discrete-time responses $y(t)$ at $t = jT$ for $T = 0.25$ sec and $j = 0, 1, \ldots, 4$ are required.

The input nomenclature follows:

The first data card:

KS—Type of problems to be solved. When KS = 1, it is the inverse Laplace transform problem. For this example, KS = 1.

N—Degree of transfer function. For this example, it is 3.

MT—Number of discrete time solutions required. In this case, it is 5.

DT—Time increment, o[illegible]npling period. For this example, it is 0.25 sec.

The second data card:

AA0, AA(1), ..., AA(N)—Coefficients of the denominator. For this example, they are 1, 3, − 10, 0.

BB(1), ..., BB(N)—Coefficients of the numerator. In this case, they are 1, 8, 12.

If the degree of the numerator is less than (N − 1), zero coefficients are assigned in the numerator. The output data of

this program are:

KS...1 N...3 MT...5 DT...0.250000

DENOMINATOR COEFFICIENT AA0,AA(1)...AA(N) ARE
0.10000000E 01 0.30000000E 01 −0.10000000E 02 0.

NUMERATOR COEFFICIENT BB(1)...BB(N) ARE
0.10000000E 01 0.80000000E 01 0.12000000E 02
INITIAL CONDITIONS X(1)...X(N) ARE
0.10000000E 01 0.50000000E 01 0.70000000E 01

THE REQUIRED SOLUTION

T	Y1	Y2	Y3	Y4
0.	0.10000E 01	0.50000E 01	0.70000E 01	
0.25000E 00	0.25897E 01	0.77179E 01	0.14744E 02	
0.50000E 00	0.51446E 01	0.12721E 02	0.25283E 02	
0.75000E 00	0.93810E 01	0.21169E 02	0.42302E 02	
0.10000E 01	0.16436E 02	0.35275E 02	0.70541E 02	

Note that $Y1 = y(t)$ and $Y2 = \dot{y}(t)$.
Example 2. Consider the following state equation:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \\ x(0) &= x_0 \end{aligned} \tag{A2}$$

where

$$A = \text{the system matrix} = \begin{bmatrix} 1 & 2 \\ 3 & -4 \end{bmatrix};\ B = \text{the input matrix} = \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix};$$

$$C = \text{the output matrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix};\ u(t) = \text{the input vector} = \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix};$$

$$x(0) = \text{the initial vector} = \begin{bmatrix} 1 \\ 1 \end{bmatrix};\ y(t) = \text{the output vector} = \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix}.$$

The input functions $u_1(t)$ and $u_2(t)$ are unit-step functions. The responses $y(t)$ at $t = jT$ for $T = 0.25$ sec and $j = 0, 1, \ldots, 4$ are required.
The input nomenclature follows:
The first data card:
KS—Type of problems to be solved. When KS = 2, it is the problem of solving linear time-invariant state equations; when KS = 3, for solving linear time-varying and nonlinear state equations. In this example KS = 2.
N—Order of the state equation. For this case, it is 2.
MT—Number of discrete-time solution required. In this example, it is 5.
DT—Time increment or sampling period. For this example, it is 0.25 sec.
The second data card:
KU—Type of input functions. If KU = 1, the input functions $u_i(t)$ are continuous-time functions. All $u_i(t)$ can be inserted *in the main program using t as an independent variable.* If KU = 2, the input functions are in the form of discrete-time input data. For illustration, in this example, KU = 2.
NU—Number of the input functions. For this example, it is 2.
NP—Number of the output functions. In this case, it is 2.
The third data card:
A(1, 1), A(1, 2), ..., A(1, N)—The entries of the first row vector in the system matrix A. For this example, they are 1, 2.
The fourth data card:
A(2, 1), A(2, 2), ..., A(2, N)—The entries of the second row vector in the same matrix. In this case, they are 3, −4.
The fifth data card:
B(1, 1), B(1, 2), ..., B(1, NU)—The entries of the first row vector in the input matrix B. For this example, they are 2, 0.
The sixth data card:
B(2, 1), B(2, 2), ..., B(2, NU)—The entries of the second row vector in the same matrix. In this case, they are 1, 1.
The seventh data card:
C(1, 1), C(1, 2), ..., C(1, N)—The entries of the first row vector in the output matrix C. In this example, they are 1, 0.
The eighth data card:
C(2, 1), C(2, 2), ..., C(2, N)—The entries of the second row vector in the same matrix. For this example, they are 0, 1.
The ninth data card:
$x(1), \ldots, x(N)$—The initial conditions. For this example, they are 1, 1.
The tenth data card:
$u(1, 1), u(1, 2), \ldots, u(1, MT)$—The discrete-time data of the first input function $u_1(t)$ evaluated at $t = jT$ for $T = 0.25$ sec and $j = 0, 1, \ldots, (MT-1)$. In this example, $u_1(t) = 1$; therefore, the discrete-time input data are 1, 1, 1, 1, 1.
The eleventh data card:
$u(2, 1), u(2, 2), \ldots, u(2, MT)$—The discrete-time data of the second input function $u_2(t)$ evaluated at $t = jT$ for $T = 0.25$ second and $j = 0, 1, \ldots, (MT-1)$. For this example, $u_2(t) = 1$; the discrete-time input data are 1,, 1, 1, 1, 1.

Since the $u_1(t)$ and $u_2(t)$ are unit-step functions (continuous-time functions), we can choose KU = 1. If we choose KU = 1, we do not need the 10th and 11th data cards. However, the statements

```
UT(1) = 1.
UT(2) = 1.
```

should be inserted in the main program.

The output data of this program are:

```
KS...2     N...2    MT...5  DT...0.250000
KU...2    NU...2    NP...2

  SYSTEM MATRIX
0.10000000E 01    0.20000000E 01
0.30000000E 01   -0.40000000E 01

  INPUT MATRIX
0.20000000E 01  0.
0.10000000E 01  0.10000000E 01

  OUTPUT MATRIX
0.10000000E 01  0.
0.              0.10000000E 01

  INITIAL CONDITIONS X(1)...X(N) ARE
0.10000000E 01  0.10000000E 01

  DISCRETE INPUT DATA
1.000   1.000   1.000   1.000   1.000
1.000   1.000   1.000   1.000   1.000

  THE REQUIRED SOLUTION

     T              Y1            Y2            Y3            Y4

0.              0.10000E 01   0.10000E 01
0.25000E 00     0.25897E 01   0.15641E 01
0.50000E 00     0.51446E 01   0.27883E 01
0.75000E 00     0.93810E 01   0.48942E 01
0.10000E 01     0.16436E 02   0.84191E 01
```

Example 3. Given a nonlinear state equation in (24) of the main paper, or

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 1 - x_1^2(t) \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t) \tag{A3}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}; \quad \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}$$

where $u(t) = 0$.

The procedures to prepare the input data cards are the same as those shown in Example 2, except the following:

(i) KS = 3.

(ii) If KU = 1 is used, the statement of the input function UT(1) = 0 is inserted in the main program as shown in the list of this program.

(iii) Any entries that consist of nonlinear or time-varying terms, or both, in the system matrix are first assumed to be zeros in the input cards, then the corresponding terms are recovered by the exact terms by substituting the state variables $x_j(t)$ by $x(j)$ and t by T in the main program. In this example, the system matrix in (A3) is first assumed as

$$A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{A4}$$

In other words, the entry $A(2, 2)$ is a nonlinear term which is assumed to be zero in (A4). The term is recovered in a statement $A(2, 2) = 1 - x(1) * x(1)$ in the main program as shown in the list of this program. The outputs of this example are plotted in Fig. 3 of the main paper. The complete computer program for solving the inverse Laplace transform, linear and nonlinear state equations follows.

Computer program

```
C      A PROGRAM FOR SOLVING INVERSE LAPLACE TRANSFORM,
C      LINEAR STATE EQUATIONS AND NONLINEAR  STATE EQUATIONS USING
C      BLOCK-PULSE FUNCTIONS.
C
C      KS=1  INVERSE LAPLACE TRANSFORM
C      KS=2  LINEAR TIME-INVARIANT STATE EQUATIONS
C      KS=3  NONLINEAR AND LINEAR TIME-VARYING STATE EQUATIONS
C
C      WHEN KS=1,
C      Y(S)= (BB(1)*S**(N-1)+...+BB(N))/(AA0*S**N+AA(1)*S**(N-1)+...
C      +AA(N))
C
C      WHEN KS=2,
C      DX(T)/DT=AX(T)+BU(T)
C      Y(T)=CX(T)
C      X(0)= INITIAL VECTOR
C
C      WHEN KS=3,
C      THE TIME-VARYING AND NONLINEAR ENTRIES IN THE MATRIX 'A' ARE
C      ASSUMED TO BE ZERO IN THE INPUT DATA CARDS,THEN THE CORRESPONDING
C      ENTRIES ARE RECOVERED BY THE EXACT TERMS IN THE MAIN PROGRAM
C      USING STATE VARIABLES X(1),...,X(N) AND TIME VARIABLE 'T' .
C
C      N= DEGREE OF THE TRANSFER FUNCTION
C       = ORDER OF THE STATE EQUATION
C      MT= NO. OF THE KNOWN DISCRETE-TIME INPUT DATA
C         =NO. OF THE OUTPUT DATA REQUIRED
C      DT= TIME INCREMENT OR THE SAMPLING PERIOD
C
C      WHEN KU=1,
C      THE INPUT FUNCTIONS UT(J) ARE CONTINUOUS-TIME FUNCTIONS.
C      WRITE ALL THE UT(J) IN THE MAIN PROGRAM USING THE INDEPENDENT
C      VARIABLE 'T'.
C
C      WHEN KU=2,
C      THE INPUT FUNCTIONS ARE DISCRETE-TIME DATA.
C
C      NU= NO. OF THE INPUT FUNCTIONS
C      NP= NO. OF THE OUTPUT FUNCTIONS
C
C
500    FORMAT(3I5,F10.6)
501    FORMAT ((5E16.8))
510    FORMAT((8F10.3))
600    FORMAT(/2X,5HKS...,I2,4X,4HN...,I2,4X,5HMT...,I3,4X,5HDT...,F10.6)
601    FORMAT(//4X, 'DENOMINATOR COEFFICIENT  AA0,AA(1)...AA(N)  ARE')
602    FORMAT((2X,6F16.8))
603    FORMAT(//4X,'NUMERATOR COEFFICIENT   BB(1)...BB(N)  ARE')
604    FORMAT(/6X,'INITIAL CONDITIONS X(1)...X(N) ARE')
605    FORMAT((2X,6F16.8))
606    FORMAT(/2X,5HKU...,I2,4X,5HNU...,I2,4X,5HNP...,I2)
607    FORMAT(//6X,'SYSTEM MATRIX')
608    FORMAT(//6X,'INPUT MATRIX')
609    FORMAT(//6X,'OUTPUT MATRIX')
610    FORMAT(//6X,'DISCRETE INPUT DATA')
611    FORMAT(//6X,'THE REQUIRED SOLUTION '/9X,'T',12X,'Y1',11X,'Y2',
      $11X,'Y3',11X,'Y4',11X,'Y5',11X,'Y6',11X,'Y7')
612    FORMAT(1X,9E13.5)
620    FORMAT((2X,8F10.3))
       DIMENSION AA(20),BB(20),A(20,20),A1(20,20),XN(20),X(20),XG(20),
      $BL(20,100),B(20,20),C(20,20),U(20,100),UL(20,100),AG(20),UT(20),
      $XK1(20),X1(20),R1(20,20),R2(20,20),XD(20,100),XD1(20),XK(20,100),
      $XD2(20),XD3(20),XC(20,100),XT(20,100),Y(20,100) ,R3(20,21)

C
C
1000   READ (5,500,END=1) KS,N,MT,DT
       WRITE(6,600)KS,N,MT,DT
       M=MT-1
       IF(KS.NE.1) GO TO 1001
       WRITE(6,601)
       READ(5,501) AA0,(AA(L),L=1,N)
       WRITE(6,602) AA0,(AA(L),L=1,N)
       WRITE(6,603)
       READ(5,501)(BB(L),L=1,N)
       WRITE(6,602)(BB(L),L=1,N)
       DO 101 L=1,N
       BB(L)=BB(L)/AA0
101    AA(L)=-AA(L)/AA0
       N1=N-1
       DO 102 L=1,N1
       L1=L+1
       DO 103 J=1,N
103    A(L,J)=0.
102    A(L,L1)=1.
       DO 104 J=1,N
       L=N+1-J
104    A(N,J)=AA(L)
       DO 110 L=1,N
       DO 120 J=1,N
120    A1(L,J)=0.
```

```
110    A1(L,L)=1.
       N2=N-1
       DO 140 L=1,N2
       S=0.
       DO 160 J=1,L
160    S=A1(L,J)*AA(J)+S
       L1=L+1
       DO 170 JJ=L1,N
       JJL=JJ-L
170    A1(JJ,JJL)=S
140    CONTINUE
       DO 180 L=1,N
       XN(L)=0.
       DO 180 J=1,N
       XN(L)=A1(L,J)*B9(J)+XN(L)
       X(L)=XN(L)
180    XG(L)=X(L)
       WRITE(5,604)
       WRITE(6,605) (X(L),L=1,N)
       DO 190 L=1,N
       DO 190 J=1,M
190    BL(L,J)=0.
       NP=N
       DO 200 L=1,NP
       DO 201 J=1,N
201    C(L,J)=0.
200    C(L,L)=1.
       GO TO 2000
1001   READ(5,500) KU,NU,NP
       WRITE(6,606) KU,NU,NP
       WRITE(6,607)
       DO 210 L=1,N
       READ(5,501)(A(L,J),J=1,N)
210    WRITE(6,602)(A(L,J),J=1,N)
       WRITE(6,608)
       DO 211 L=1,N
       READ(5,501)(B(L,J),J=1,NU)
211    WRITE(6,602)(B(L,J),J=1,NU)
       WRITE(6,609)
       DO 212 L=1,NP
       READ(5,501)(C(L,J),J=1,N)
212    WRITE(6,602)(C(L,J),J=1,N)
       WRITE(6,604)
       READ(5,501)(X(J),J=1,N)
       WRITE(6,602)(X(J),J=1,N)
       DO 214 L=1,N
       XN(L)=X(L)
214    XG(L)=X(L)
       IF(KU.EQ.2) GO TO 251
       T=0.
       DO 260 J=1,MT
       DO 261 L=1,NU
C
C      *** IF KU=1, INSERT ALL CONTINUOUS-TIME INPUT FUNCTIONS
C      UT(1),... ,UT(NU) HERE AND USE 'T' AS AN INDEPENDENT VARIABLE.***
C
       UT(1)=0.
C
C      ***
C
261    U(L,J)=UT(L)
260    T=T+DT
       GO TO 252
251    WRITE(6,610)
       DO 213 L=1,NU
       READ(5,510)(U(L,J),J=1,MT)
213    WRITE(6,620)(U(L,J),J=1,MT)
252    DO 215 L=1,NU
       DO 215 J=1,M
       J1=J+1
215    UL(L,J)=(U(L,J)+U(L,J1))/2.
       DO 700 KL=1,N
       DO 700 KJ=1,M
       S=0.
       DO 701 KK=1,NU
       S=S+B(KL,KK)*UL(KK,KJ)
 700   BL(KL,KJ)=S
       IF(KS.EQ.2) GO TO 2000
       K=0
       T=0.
3000   CONTINUE
C
C      *** IF KS=3, REPLACE THE ENTRIES THAT ARE NONLINEAR AND / OR TIME
C      VARYING TERMS IN THE MATRIX 'A' BY THE EXACT TERMS USING THE STATE
C      VARIABLES X(1), ... ,X(N) AND THE TIME VARIABLE 'T'.***
C
       A(2,2)=1.-X(1)*X(1)
C
C      ***
C
       T=T+DT
       K=K+1
```

```
2000  CALL MALTP(N,N,A,1,XG,AG)
      DO 216 L=1,N
      DO 217 J=1,N
217   A1(L,J)=-A(L,J)/2.
216   A1(L,L)=1./DT+A1(L,L)
      CALL DINVM (A1,R3,N)
      DO 705 L=1,N
      DO 705 J=1,N
705   R1(L,J)=R3(L,J)
      IF (KS.NE.3) GO TO 4000
      DO 218 L=1,N
218   XK1(L)=(AG(L)+BL(L,K))/2.
      CALL MALTP(N,N,R1,1,XK1,X1)
      DO 219 L=1,N
      XD3(L)=X1(L)+XG(L)
      X(L)=-XG(L)+2.*XD3(L)
      XG(L)=X(L)
219   XC(L,K)=XD3(L)
      IF (K.EQ.M) GO TO 5000
      GO TO 3000
4000  DO 710 KL=1,N
      DO 710 KJ=1,N
      S=0.
      DO 710 KK=1,N
      S=S+R1(KL,KK)*A(KK,KJ)
710   R2(KL,KJ)=S
      DO 230 L=1,N
      DO 230 J=1,M
230   XK(L,J)=AG(L)+BL(L,J)
      DO 231 J=1,M
      IF (J.GT.1) GO TO 301
      DO 232 L=1,N
232   XK1(L)=XK(L,J)/DT
      CALL MALTP(N,N,R1,1,XK1,XD1)
      DO 233 L=1,N
233   XD(L,J)=XD1(L)
      GO TO 231
301   J2=J-1
      DO 234 L=1,N
234   XK1(L)=XK(L,J)/DT-XK(L,J2)/DT
      CALL MALTP(N,N,R1,1,XK1,XD2)
      CALL MALTP(N,N,R2,1,XD1,XD3)
      DO 235 L=1,N
      XD(L,J)=XD(L,J2)+XD3(L)+XD2(L)
235   XD1(L)=XD(L,J)
231   CONTINUE
      DO 240 J=1,M
      IF (J.GT.1) GO TO 302
      DO 241 L=1,N
241   XC(L,J)=XD(L,J)*DT/2.+XG(L)
      GO TO 240
302   J2=J-1
      DO 242 L=1,N
242   XC(L,J)=XC(L,J2)+(XD(L,J2)+XD(L,J))*DT/2.
240   CONTINUE
5000  DO 400 L=1,N
400   XT(L,1)=XN(L)
      DO 401 L=1,N
      DO 401 J=2,MT
      J2=J-1
401   XT(L,J)=-XT(L,J2)+2.*XC(L,J2)
      DO 720 KL=1,NP
      DO 720 KJ=1,MT
      S=0.
      DO 720 KK=1,N
      S=S+C(KL,KK)*XT(KK,KJ)
720   Y(KL,KJ)=S
      WRITE (6,611)
      DO 402 J=1,MT
      T=(J-1)*DT
402   WRITE(6,612) T,(Y(L,J),L=1,NP)
      GO TO 1000
1     STOP
      END

      SUBROUTINE  MALTP(N,M,A,K,B,C)
      DIMENSION A(20,20),B(20),C(20)
      LK=K
      DO 10 L=1,N
      S=0.
      DO 10 J=1,M
      S=S+A(L,J)*B(J)
10    C(L)=S
      RETURN
      END

      SUBROUTINE DINVM (A,D,N)
      DIMENSION A(20,20),D(20,21)
      DO 10 I=1,N
      DO 10 J=1,N
```

```
10    D(I,J)=A(I,J)
      M=N+1
      KK=0
      JJ=0
      DO 20 K=1,N
      DO 30 J=1,N
30    D(J,M)=0.
      D(K,M)=1.
      JJ=KK+1
      LL=JJ
      KK=KK+1
40    IF (ABS(D(JJ,KK))-1.E-4)50,50,60
50    JJ=JJ+1
      GO TO 40
60    IF (LL-JJ)70,80,70
70    DO 90 MM=1,M
      DTEMP=D(LL,MM)
      D(LL,MM)=D(JJ,MM)
90    D(JJ,MM)=DTEMP
80    DIV=D(K,K)
      DO 100 LJ=1,M
      J=M+1-LJ
100   D(K,J)=D(K,J)/DIV
      DO 110 I=1,N
      FAC=D(I,K)
      DO 110 LJ=1,M
      J=M+1-LJ
      IF (I-K)120,110,120
120   D(I,J)=D(I,J)-FAC*D(K,J)
110   CONTINUE
      DO 130 J=1,N
130   D(J,K)=D(J,M)
20    CONTINUE
      RETURN
      END
```

PROCEEDINGS

OF THE INSTITUTION OF ELECTRICAL ENGINEERS

Control & Science

Synthesis of optimal block controllers for multivariable control systems and its inverse optimal-control problem

Y.J. Wei, M.Sc., and Prof. L.S. Shieh, M.Sc., Ph.D.

Indexing terms: *Multivariable control systems, Control-system synthesis, Optimal control*

Abstract

A new method is presented to synthesise optimal block controllers for a class of multivariable control systems represented by the block companion form. The reverse process of obtaining the optimal block controller is used to determine the block-weighting matrices of the quadratic performance index from prescribed control specifications.

1 Introduction

The accurate description of linear time-invariant systems in the time domain may result in m nth-degree coupled differential equations, or an nth-degree matrix differential equation with $m \times m$ matrix coefficients[1] as

$$\sum_{i=1}^{n+1} A_i D^{i-1} x = u \tag{1a}$$

$$y = \sum_{i=1}^{n} C_i D^{i-1} x \tag{1b}$$

nd

$$D^{i-1}x(0) = \alpha_i, \quad i = 1, 2, \ldots, n \tag{1c}$$

where y is an $m \times 1$ output vector, u is an $m \times 1$ input vector and x is an $m \times 1$ state vector. A_i and C_i are $m \times m$ matrix coefficients, and the differential operator $D = d/dt$. When each initial vector α_i is an $m \times 1$ null vector, the corresponding frequency-domain representation of eqn. 1 is an nth-degree matrix transfer function written as

$$Y(s) = T(s)U(s) \tag{2a}$$

where $Y(s)$ and $U(s)$ are the $m \times 1$ output vector and the $m \times 1$ input vector, respectively, and the matrix transfer function $T(s)$ is

$$T(s) = N_r(s)D_r^{-1}(s) = D_l^{-1}(s)N_l(s) \tag{2b}$$

The matrix polynomials $D_r(s)$ and $N_r(s)$ with appropriate size are right coprime, $D_l(s)$ and $N_l(s)$ left coprime. Let us define

$$D_r(s) = I_m s^n + A_n s^{n-1} + \ldots + A_2 s + A_1 \tag{3}$$

$$N_r(s) = C_n s^{n-1} + C_{n-1} s^{n-2} + \ldots + C_2 s + C_1$$

where A_i and C_i are $m \times m$ constant matrices. The corresponding first-degree state equation in the controllable phase-variable block form or in the controllable block companion form is

$$\dot{X} = AX + Bu \tag{4a}$$

$$y = CX; x(0) = X_0 \tag{4b}$$

where

$$A = \begin{bmatrix} O_m & I_m & O_m & \cdot & O_m \\ O_m & O_m & I_m & \cdot & O_m \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ -A_1 & -A_2 & -A_3 & \cdot & -A_n \end{bmatrix}, \quad B = \begin{bmatrix} O_m \\ O_m \\ \cdot \\ I_m \end{bmatrix}, \quad X = \begin{bmatrix} X_1 \\ X_2 \\ \cdot \\ X_n \end{bmatrix} \tag{4c}$$

$$C = [C_1 \quad C_2 \quad \ldots \quad C_n] \tag{4d}$$

The block elements A_i, O_m, I_m and C_i are $m \times m$ constant matrices, $m \times m$ null matrix, $m \times m$ identity matrix and $m \times m$ constant matrices, respectively. The vector X consists of n blocks (X_i, $i = 1, 2, \ldots, n$) and each $m \times 1$ block X_i consists of m state variables. In this paper, we define the vector X as a block vector. Because the state equation in eqn. 4 is formulated in the phase-variable block form, the X is defined as a vector in the phase-variable block co-ordinate. As a result, the $X(0)$ is an initial block vector. From a conventional viewpoint, the same vector X is viewed as a vector with nm state variables in a general co-ordinate. Therefore, the same state equation in eqn. 4 is viewed as a state equation in a general co-ordinate. In this paper, all the derivations are based on the state equation in the phase-variable block co-ordinate rather than a general co-ordinate.

The objectives of this paper are described as follows:

(*a*) Obtain the optimal block-control law $u = -R^{-1}B^TPX = -KX$ (where the feedback-gain matrix $K = R^{-1}B^TP$ consists of $m \times m$ block elements K_i, $i = 1, \ldots, n$) to minimise the quadratic performance index

$$J = \frac{1}{2}\int_0^{\infty} [X^TQX + u^TRu]dt \tag{5a}$$

for the dynamic system formulated in the phase-variable block co-ordinate in eqn. 4. The T designates transpose, the weighting matrix R is an assigned $m \times m$ positive-definite matrix, and the block-weighting matrix Q is an assigned $nm \times nm$ nonnegative definite-symmetric matrix with $m \times m$ block elements $Q_{i,j} = Q_{j,i}^T$, or

$$Q = \begin{bmatrix} Q_{11} & Q_{12} & \cdots & Q_{1n} \\ Q_{21} & Q_{22} & \cdots & Q_{2n} \\ \cdot & \cdot & \cdots & \cdot \\ Q_{n1} & Q_{n2} & \cdots & Q_{nn} \end{bmatrix} = Q^T \tag{5b}$$

The $nm \times nm$ matrix P is the positive-definite solution of the steady-state Riccati equation[2]

$$PA + A^TP + Q - PBR^{-1}B^TP = O_{nm} \tag{5c}$$

The same P can be also solved from the following canonical form:[2]

$$\begin{bmatrix} \dot{X} \\ \dot{G} \end{bmatrix} = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} X \\ G \end{bmatrix},$$

$$G(\infty) = PX(\infty) = O_{nm \times 1}, X(0) = X_0 \tag{5d}$$

Paper 8267C, received 3rd November 1978 and in revised form 24th January 1979

Mr. Wei and Prof. Shieh are with the Department of Electrical Engineering, University of Houston, Central Campus, Houston, Texas 77004, USA

0020-3270/79/050449 + 08 $01·50/0

It is noted that, if the pair $[A, B]$ is controllable and the pair $[A, L]$ is observable (where $Q = LL^T$), then the closed-loop system is not only optimal but stable.

(b) Determine the block-weighting matrices Q and R of the quadratic performance index in eqn. 5a if the optimal block controller K is assigned or if the closed-loop poles (or the equivalent control specifications[3]) of the optimal controlled system are prescribed.

2 Linear optimal-block-regulator problem

In the conventional synthesis of the linear-regulator problem, the state equation in eqn. 4 is viewed as a state equation in a general co-ordinate. An optimal control law is then derived by solving eqns. 5c or 5d. In this paper, the state equation in eqn. 4 is considered as a state equation in the phase-variable block co-ordinate. The optimal-block-control law is derived as follows.

Expanding eqn. 4 and adding a trivial identity yields

$$X_1 = X_1$$
$$\dot{X}_1 = X_2$$
$$\ddot{X}_1 = X_3 = \dot{X}_2$$
$$\cdots\cdots$$
$$X^{(n)} = X_n = -A_1X_1 - A_2X_2 - \ldots - A_nX_n + u \qquad (6a)$$

Rewriting the last equation in eqn. 6a gives

$$u = A_1X_1 + A_2\dot{X}_1 + \ldots + A_nX_1^{(n-1)} + X_1^{(n)} \qquad (6b)$$

Substituting eqn. 6 into eqn. 5a, we have an alternate form of the cost function as

$$F(X, u) = F(X_1, \dot{X}_1, \ldots, X_1^{(n)}) = F(X^*) = \tfrac{1}{2}X^{*T}Q^*X^* \qquad (7)$$

where

$$Q^* = \begin{bmatrix} Q^*_{11} & Q^*_{12} & \cdots & Q^*_{1n} & A_1^TR \\ Q^*_{21} & Q^*_{22} & \cdots & Q^*_{2n} & A_2^TR \\ & & \cdots & & \\ Q^*_{n1} & Q^*_{n2} & \cdots & Q^*_{nn} & A_n^TR \\ RA_1 & RA_2 & \cdots & RA_n & R \end{bmatrix} = Q^{*T}, X^* = \begin{bmatrix} X_1 \\ \dot{X}_1 \\ \vdots \\ X_1^{(n-1)} \\ X_1^{(n)} \end{bmatrix}$$

$$Q^*_{i,j} = Q_{i,j} + A_i^TRA_j = Q^{*T}_{j,i}$$

The $(n+1)m \times (n+1)m$ constant matrix Q^* is a block weighting matrix with $m \times m$ block elements. Applying the gradient matrix operations[4] to the quadratic cost function in eqn. 7 yields

$$F_{X_1} = [I_m \quad O_m \quad \cdots \quad O_m]Q^*X^*$$

$$\frac{d}{dt}F_{\dot{X}_1} = [O_m \quad I_m \quad \cdots \quad O_m]Q^*\dot{X}^*$$

$$\cdots\cdots$$

$$\frac{d^n}{dt^n}F_{X_1^{(n)}} = [O_m \quad O_m \quad \cdots \quad I_m]Q^*X^{*(n)} \qquad (8)$$

Substituting eqn. 8 into the following Euler's equation[5]

$$F_{X_1} - \frac{d}{dt}F_{\dot{X}_1} + \frac{d^2}{dt^2}F_{\ddot{X}_1} - \ldots + (-1)^n\frac{d^n}{dt^n}F_{X_1^{(n)}} = O_{m\times 1} \qquad (9)$$

we have

$$D_1X_1 + D_2\dot{X}_1 + D_3X_1^{(2)} + \ldots + D_{2n+1}X_1^{(2n)} = O_{m\times 1} \qquad (10a)$$

where

$$[D_1 \quad D_2 \quad D_3 \quad \ldots \quad D_{2n+1}] = [I_m \quad -I_m \quad I_m \quad \ldots \quad (-1)^nI_m] \times$$

$$\begin{bmatrix} Q^*_{11} & Q^*_{12} & Q^*_{13} & \cdots & Q^*_{1n} & A_1^TR & O_m & \cdots & O_m & O_m & O_m \\ O_m & Q^*_{21} & Q^*_{22} & \cdots & Q^*_{2,n-1} & Q^*_{2,n} & A_2^TR & \cdots & O_m & O_m & O_m \\ O_m & O_m & Q^*_{31} & \cdots & Q^*_{3,n-2} & Q^*_{3,n-1} & Q^*_{3,n} & \cdots & O_m & O_m & O_m \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot & \cdot & \cdots & \cdot & \cdot & \cdot \\ O_m & O_m & O_m & \cdots & Q^*_{n,1} & Q^*_{n,2} & Q^*_{n,3} & \cdots & Q^*_{n,n} & A_n^TR & O_m \\ O_m & O_m & O_m & \cdots & O_m & RA_1 & RA_2 & \cdots & RA_{n-1} & RA_n & R \end{bmatrix} \qquad (10b)$$

Expanding eqn. 10b we have

$$D_1 = Q^*_{11} = Q_{11} + A_1^TRA_1$$
$$D_2 = Q^*_{12} - Q^*_{21} = Q_{12} + A_1^TRA_2 - Q_{21} - A_2^TRA_1$$
$$\cdots\cdots$$
$$D_{2n+1} = R \qquad (10c)$$

Taking the Laplace transform of eqn. 10a and neglecting the initial conditions we have the matrix polynomial $D(s)$:

$$D(s)X_1(s) = [D_{2n+1}s^{2n} + D_{2n}s^{2n-1} + \ldots$$
$$+ D_2s + D_1]X_1(s) = O_{m\times 1} \qquad (11)$$

where $D_{2k+1} = D^T_{2k+1}$, $k = 0, 1, \ldots, n$ and $D_{2k} = -D^T_{2k}$, $k = 1, 2, \ldots, n$. It is well known that the poles of the state equation in eqn. 5d are symmetrically distributed about the origin in the s-plane, so are the roots of the determinant of the matrix polynomial $D(s)$ in eqn. 11. Performing the spectral factorisation[6,7] of the matrix polynomial $D(s)$ results in a stable matrix polynomial $\mathbf{\Delta}(s)$ and an unstable matrix polynomial $\mathbf{\Delta}(-s)$, i.e.

$$D(s) = F^T\mathbf{\Delta}(-s)^T\mathbf{\Delta}(s)F \qquad (12)$$

where

$$R = F^TF = D_{2n+1}$$

and

$$\mathbf{\Delta}(s) = I_ms^n + E_ns^{n-1} + \ldots + E_2s + E_1$$

The required optimal-block-control law is then obtained from eqns. 6b and 12 as

$$u = [K_1 \quad K_2 \ldots K_n]X \qquad (13)$$

where

$$K_i = A_i - E_i, \quad i = 1, 2, \ldots, n$$

When the given system is not in a phase-variable block form, a newly developed algorithm shown in Appendix 8 can be applied to obtain a block linear transformation that transforms a class of state equations in a general co-ordinate into the phase-variable block co-ordinate. Thus the proposed method can be applied to determine the optimal block controller.

3 Inverse optimal control problem

Given a set of prescribed closed-loop poles, or equivalent control specifications,[3] we wish to determine the weighting matrices Q and R of the quadratic performance index in eqn. 5a by which the controlled feedback system has prescribed closed-loop poles and the feedback-control law is optimal. This is an inverse optimal-control problem. Kalman[8] initiated the inverse problem for a linear time-invariant single-input system. Chang,[9] Tyler and Tuteur[10] have studied the problem via the root-locus method, while Molinari,[11] and Anderson and Shannon[12] have investigated the problem for a multivariable system. All the developed methods are based on the system equation formulated in a general co-ordinate rather than in a phase-variable block co-ordinate. Since the multivariable dynamic system is formulated in a matrix differential equation, it is more natural to investigate the problem in the phase-variable block co-ordinate than that in the general co-ordinate.

It is well known that a feedback-gain matrix can always be obtained to give a system with prescribed closed-loop poles if a system is controllable. However, the feedback controller may not be optimal. In this paper we determine the block-weighting matrices Q and R of the quadratic performance index by which the feedback controller not only provides the controlled system with prescribed closed-loop poles but also performs optimally. The steps involved are described as follows:

Step 1

Define a characteristic matrix polynomial $\mathbf{\Delta}(s)$ of the desired closed-loop system whose matrix coefficients consist of some unknown parameters (for example, the damping ratio ξ and the undamped natural angular frequency ω_n etc.) to be adjusted. The $\mathbf{\Delta}(s)$ is

$$\mathbf{\Delta}(s) = I_ms^n + E_ns^{n-1} + \ldots + E_2s + E_1 \qquad (14a)$$

If the desired characteristic polynomial of the closed-loop system is

$$[d(s)]^m = (s^n + d_ns^{n-1} + \ldots + d_2s + d_1)^m \qquad (14b)$$

where $d(s)$ is a polynomial whose coefficients consist of adjustable parameters. The characteristic matrix polynomial becomes

$$\Delta(s) = d(s)I_m = I_m s^n + d_n I_m s^{n-1} + \ldots + d_2 I_m s + d_1 I_m \qquad (14c)$$

where

$$E_i = d_i I_m$$

Step 2
Construct a matrix polynomial $D(s)$ using $\Delta(s)$ in eqn. 14

$$\begin{aligned} D(s) &= D_{2n+1}s^{2n} + D_{2n}s^{2n-1} + \ldots + D_2 s + D_1 \\ &= F^T\Delta^T(-s)\Delta(s)F \\ &= F^T[I_m s^{2n} + (E_n - E_n^T)s^{2n-1} \\ &\quad + (E_{n-1} - E_n^T E_n + E_{n-1}^T)s^{2n-2} + \ldots + E_1^T E_1]F \end{aligned} \qquad (15)$$

where $D_{2n+1} = F^T F = R$ is a weighting matrix to be determined.

Step 3
Solve the block weighting matrices Q and R from eqns. 10 and 15 in terms of adjustable parameters, or

$$\begin{aligned} D_{2n+1} &= F^T F \\ D_{2n} &= RA_n - A_n^T R = F^T(E_n - E_n^T)F \\ &\ldots\ldots \\ D_2 &= Q_{12} + A_1^T RA_2 - Q_{21} - A_2^T RA_1 \\ &= F^T(E_1^T E_2 - E_2^T E_1)F \\ D_1 &= Q_{11} + A_1^T RA_1 = F^T E_1^T E_1 F \end{aligned} \qquad (16)$$

Step 4
Determine the required block weighting matrices Q and R by adjusting the assigned unknown parameters such that R is positive definite and Q is nonnegative definite symmetric.

The procedures can be well illustrated by the following gas-turbine example.

4 An illustrative example

Consider the following linearised two-shaft gas-turbine model:[13-15]

$$\begin{bmatrix} \dot z_1 \\ \dot z_2 \\ \dot z_3 \\ \dot z_4 \end{bmatrix} = \begin{bmatrix} -1\cdot268 & -0\cdot04528 & 1\cdot498 & 951\cdot5 \\ 1\cdot002 & -1\cdot957 & 8\cdot52 & 1240 \\ 0 & 0 & -10 & 0 \\ 0 & 0 & 0 & -100 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 10 & 0 \\ 0 & 100 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \qquad (17a)$$

and

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} \qquad (17b)$$

The state equation in eqn. 17 is a system formulated in a general co-ordinate. To apply the proposed method, a block-linear-transformation matrix T is determined from the newly developed method shown in Appendix 8. The block linear transformation is

$$z = TX \qquad (18)$$

where

$$T = \left[\begin{array}{cc|cc} 14\cdot98 & 95150 & 0 & 0 \\ 85\cdot2 & 124000 & 0 & 0 \\ \hline 18\cdot5671 & -2622\cdot1 & 10 & 0 \\ -0\cdot005214 & 136\cdot829 & 0 & 100 \end{array}\right]$$

and X is in the phase-variable block co-ordinate and consists of two block vectors (X_i, $i = 1, 2$) and each vector X_i consists of two state variables ($x_{i,j}$, $j = 1, 2$, $i = 1, 2$). The state equation in the phase-variable block co-ordinate is

$$\begin{bmatrix} \dot x_{1,1} \\ \dot x_{1,2} \\ \hline \dot x_{2,1} \\ \dot x_{2,2} \end{bmatrix} = \left[\begin{array}{cc|cc} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \hline -18\cdot5671 & 2622\cdot1 & -11\cdot8567 & 262.21 \\ 0\cdot005214 & -136\cdot829 & 0 & -101\cdot368 \end{array}\right] \begin{bmatrix} x_{1,1} \\ x_{1,2} \\ \hline x_{2,1} \\ x_{2,2} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \hline 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \qquad (19a)$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 14\cdot98 & 95150 & 0 & 0 \\ 85\cdot2 & 124000 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{1,1} \\ x_{1,2} \\ \hline x_{2,1} \\ x_{2,2} \end{bmatrix} \qquad (19b)$$

where

$$A_1 = \begin{bmatrix} 18\cdot5671 & -2622\cdot1 \\ -0\cdot005214 & 136\cdot829 \end{bmatrix}, A_2 = \begin{bmatrix} 11\cdot8567 & -262\cdot21 \\ 0 & 101\cdot368 \end{bmatrix} \qquad (19c)$$

$$C_1 = \begin{bmatrix} 14\cdot98 & 95150 \\ 85\cdot2 & 124000 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad (19d)$$

It is required to determine two optimal block controllers for the gas-turbine system by using

(*a*) assigned weighting matrices Q and R of the quadratic performance index
(*b*) assigned control specifications.

The procedures are described as follows:

(*a*) *Optimal-block-controller design via assigned weighting matrices*
The cost function of the state equation in the original co-ordinate in eqn. 17 is

$$J = \frac{1}{2}\int_0^\infty [z^T\bar{Q}z + u^T Ru]\,dt \qquad (20)$$

where $Q = I_4$ and $R = I_2$ that were suggested by Tiwari *et al.*[15] The corresponding cost function in the phase-variable block co-ordinate is

$$J = \frac{1}{2}\int_0^\infty [X^T QX + u^T Ru]\,dt \qquad (21)$$

where $R = I_2$ and

$$Q = T^T\bar{Q}T = \left[\begin{array}{c|c} Q_{11} & Q_{12} \\ \hline Q_{21} & Q_{22} \end{array}\right]$$

$$= \left[\begin{array}{cc|cc} 7828\cdot1776 & 11941461\cdot5 & 185\cdot671 & -\cdot521389 \\ 11941461\cdot5 & 24436416630 & -26221 & 13682\cdot9 \\ \hline 185\cdot671 & -26221 & 100 & 0 \\ -\cdot521389 & 13682\cdot9 & 0 & 10000 \end{array}\right]$$

From eqn. 10 we have

$$[D_1 \; D_2 \ldots D_5] = [I_2 \; -I_2 \; I_2] \begin{bmatrix} Q_{11}^* & Q_{12}^* & A_1^T R & O_2 & O_2 \\ O_2 & Q_{21}^* & Q_{22}^* & A_2^T R & O_2 \\ O_2 & O_2 & RA_1 & RA_2 & R \end{bmatrix} \qquad (22)$$

By expanding eqn. 22, $D(s)$ in eqn. 11 becomes

$$
\begin{aligned}
D(s) &= D_5 s^4 + D_4 s^3 + \ldots + D_1 \\
&= Rs^4 + (RA_2 - A_2^T R)s^3 \\
&\quad + (RA_1 + A_1^T R - Q_{22} - A_2^T RA_2)s^2 \\
&\quad + (Q_{12} + A_1^T RA_2 - Q_{21} - A_2^T RA_1)s \\
&\quad + (Q_{11} + A_1^T RA_1) = O_2
\end{aligned} \tag{23}
$$

where

$$
D_5 = I_2, D_4 = \begin{bmatrix} 0 & -262.21 \\ 262.21 & 0 \end{bmatrix}
$$

$$
D_3 = \begin{bmatrix} -2.03447 & 486.84 \\ 486.84 & -88755.96 \end{bmatrix}
$$

$$
D_2 = \begin{bmatrix} 0 & 52440.924 \\ -52440.924 & 0 \end{bmatrix}
$$

$$
D_1 = \begin{bmatrix} 8172.92 & 11892776 \\ 11892776 & 2.44433 \times 10^{10} \end{bmatrix}
$$

Performing the spectral factorisation[7] on the $D(s)$ gives

$$
\Delta(s) = I_2 s^2 + E_2 s + E_1 \tag{24}
$$

where

$$
E_2 = \begin{bmatrix} 17.2396 & -261.906 \\ 0.30451 & 576.845 \end{bmatrix} \text{ and } E_1 = \begin{bmatrix} 46.925 & -3929.922 \\ 77.27215 & 156294.2 \end{bmatrix}
$$

From eqn. 13 we have the optimal block controllers in the block co-ordinate and original co-ordinate as

$$
\begin{aligned}
u &= [A_1 - E_1 \,\vdots\, A_2 - E_2] X \\
&= -\left[\begin{array}{cc:cc} 28.3576 & -1307.82 & 5.3829 & 0.3045 \\ 77.2774 & 156157.4 & 0.304513 & 475.476 \end{array}\right] X
\end{aligned} \tag{25a}
$$

$$
\begin{aligned}
&= [A_1 - E_1 \,\vdots\, A_2 - E_2] T^{-1} z \\
&= \left[\begin{array}{cc:cc} -0.36296 & 0.279346 & 0.53829 & 0.003045 \\ 0.598572 & 0.795425 & 0.0304513 & 4.75476 \end{array}\right] z
\end{aligned} \tag{25b}
$$

(*b*) *The optimal-block-controller design via assigned control specifications*

The design goals are specified as follows:

(i) static decoupling
(ii) final values of the unit-step responses are unity
(iii) peak time t_p that is the time required for the unit-step response to reach the first peak of the overshoot is near 0·01 s
(iv) maximum percentage overshoot is less than 10%.

To reach the first design goal, the characteristic matrix polynomial is defined as

$$
\Delta(s) = I_2 s^2 + E_2 s + E_1 \tag{26}
$$

where

$$
E_2 = \begin{bmatrix} 2\xi\omega_n & 0 \\ 0 & 2\xi\omega_n \end{bmatrix} \text{ and } E_1 = \begin{bmatrix} \omega_n^2 & 0 \\ 0 & \omega_n^2 \end{bmatrix}
$$

ξ (damping ratio) and ω_n (undamped natural frequency) are unknown parameters to be determined. To satisfy the third design goal we can estimate ω_n from the following rule of thumb in designs[16] as

$$
\omega_n \simeq \frac{\pi}{t_p} \simeq \frac{3.14}{0.01} \simeq 300 \text{ rad/s} \tag{27a}
$$

Also, from another rule of thumb,[16] we can estimate ξ to meet the fourth design goal as

$$
\xi \cong -\frac{\ln M_p}{\pi} = -\frac{\ln 0.1}{3.14} \simeq 0.75 \tag{27b}
$$

The choices in eqn. 27 imply that the closed-loop poles have been assigned at

$$
s_{1,2} = -\xi\omega_n \pm j\omega_n\sqrt{1-\xi^2} \simeq -225 \pm j\,198.43 \tag{27c}
$$

From eqn. 26 $D(s)$ can be determined as

$$
\begin{aligned}
D(s) &= F^T \Delta^T(-s) \Delta(s) F \\
&= F^T F s^4 + (2\omega_n^2 - 4\xi^2\omega_n^2) F^T F s^2 + \omega_n^4 F^T F \\
&= Rs^4 + (RA_2 - A_2^T R)s^3 \\
&\quad + (RA_1 + A_1^T R - Q_{22} - A_2^T RA_2)s^2 \\
&\quad + (Q_{12} + A_1^T RA_2 - Q_{12}^T - A_2^T RA_1)s \\
&\quad + (Q_{11} + A_1^T RA_1)
\end{aligned} \tag{28}
$$

For simplicity, let $Q_{12} = Q_{21} = O_2$. Equating the matrix coefficients of the same power of eqn. 28, we obtain the following matrix equations:

$$
(a)\ R = F^T F \tag{29a}
$$

$$
(b)\ RA_2 - A_2^T R = O_2 \tag{29b}
$$

$$
(c)\ RA_1 + A_1^T R - Q_{22} - A_2^T RA_2 = (2\omega_n^2 - 4\xi^2\omega_n^2) F^T F \tag{29c}
$$

$$
(d)\ A_1^T RA_2 - A_2^T RA_1 = O_2 \tag{29d}
$$

$$
(e)\ Q_{11} + A_1^T RA_1 = \omega_n^4 F^T F \tag{29e}
$$

R is an $m \times m$ symmetric and positive-definite matrix which has $m(m+1)/2$ unknown elements to be determined. The left-hand-side matrices in eqns. 29*b* and 29*d* are skew-symmetric matrices. Expanding the matrix equations in eqns. 29*b* and 29*d* results in $m(m-1)$ simultaneous equations with $m(m+1)/2$ unknown variables in R. In general, there are an infinite number of solutions. However, if k independent simultaneous equations exist, and $k < m(m+1)/2$, then we can assume $[m(m+1)/2 - k]$ constants to solve k unknown variables in R. The choice of the assigned constants in R is a design freedom and a certain amount of experience is helpful. In this example, we assume R_{11}, which is the first leading diagonal element, is unity. Thus we can solve for R and F in eqn. 29*a* as

$$
R = \begin{bmatrix} 1 & 2.92934 \\ 2.92934 & 51058.01562 \end{bmatrix} = F^T F \tag{30}
$$

where

$$
F = \begin{bmatrix} 0.999916 & 5.737 \times 10^{-5} \\ 0.0129466 & 225.96 \end{bmatrix}
$$

Note that R is a positive-definite matrix. From eqns. 30, 29*c* and 29*e* we can solve for Q_{11} and Q_{22} as

$$
Q_{11} = \begin{bmatrix} \omega_n^4 - 345.55808 & 2.929341\omega_n^4 + 77629.05 \\ 2.929341\omega_n^4 + 77629.05 & 51058.0156\,\omega_n^4 - 7.6070451 \times 10^8 \end{bmatrix} \tag{31a}
$$

$$
Q_{22} = \begin{bmatrix} 4\xi^2\omega_n^2 - 2\omega_n^2 - 140.5813 & 2.929341(4\xi^2\omega_n^2 - 2\omega_n^2) - 2844.916 \\ 2.929341(4\xi^2\omega_n^2 - 2\omega_n^2) - 2844.916 & 51058.0156(4\xi^2\omega_n^2 - 2\omega_n^2) - 524561317.4 \end{bmatrix} \tag{31b}
$$

Substituting $\omega_n = 300$ and $\xi = 0.75$ into eqn. 31 yields positive-definite matrices Q_{11} and Q_{22}. Thus the optimal block controllers can be easily found in the block co-ordinate and in the original co-ordinate as

$$
\begin{aligned}
u &= [A_1 - E_1 \,\vdots\, A_2 - E_2] X \\
&= -\left[\begin{array}{cc:cc} 89981.4329 & 2622.1 & 438.1433 & 262.21 \\ 0.005214 & 89863.17 & 0 & 348.632 \end{array}\right] X
\end{aligned} \tag{32a}
$$

$$= [A_1 - E_1 \,\vdots\, A_2 - E_2]\,T^{-1}z$$

$$= \left[\begin{array}{cc|cc} -1767{\cdot}7 & 1357{\cdot}37 & 43{\cdot}8143 & 2{\cdot}6221 \\ 1{\cdot}2182 & -0{\cdot}21391 & 0 & 3{\cdot}48632 \end{array}\right] z \qquad (32b)$$

The block-weighting matrix Q in the block co-ordinate and the weighting matrix $\bar{Q}$ in the original co-ordinate are

$$Q = \left[\begin{array}{cc|cc} 8{\cdot}1\times10^{9} & 2{\cdot}37277\times10^{10} & 0 & 0 \\ 2{\cdot}37277\times10^{10} & 4{\cdot}13569\times10^{14} & 0 & 0 \\ \hline 0 & 0 & 11703{\cdot}42 & 31850{\cdot}2 \\ 0 & 0 & 31850{\cdot}2 & 80169820 \end{array}\right] \qquad (33a)$$

and

$$\bar{Q} = \left[\begin{array}{cc|cc} 3253160 & -2454610 & 47{\cdot}2383 & -2{\cdot}91217 \\ -2454610 & 1878440 & -33{\cdot}808 & -5{\cdot}9383 \\ \hline 47{\cdot}2383 & -33{\cdot}808 & 117{\cdot}0342 & 31{\cdot}8502 \\ -2{\cdot}91217 & -5{\cdot}9383 & 31{\cdot}8502 & 8016{\cdot}982 \end{array}\right] \qquad (33b)$$

It is noticed that[2] any arbitrarily prescribed closed-loop poles or control specifications may not result in a positive-definite matrix R and nonnegative-definite matrix Q. The constraints suggested by Anderson[2] should be satisfied. In addition, some realistic constraints to the amplitudes of the control signals, for example the limitations of the actuator amplitude and rate change of amplitude, should be also examined.

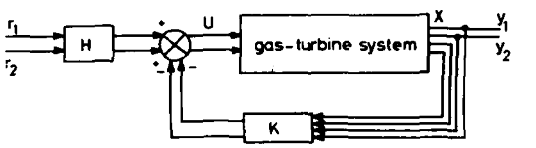


Fig. 1
Structure of designed system

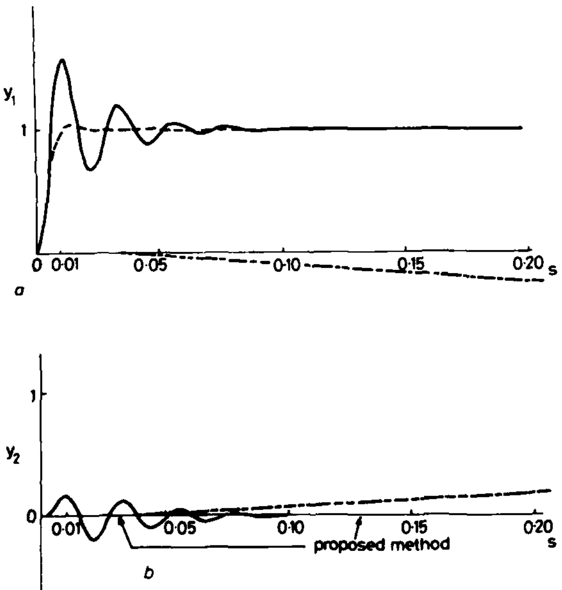


Fig. 2
Responses of various designed systems to a unit step in r_1

$r_1 = 1$
$r_2 = 0$
——— McMorran's method:
— — — proposed method: $\xi = 0{\cdot}75$; $\omega_n = 300$
-- - - proposed method: $\bar{Q} = I_4$; $R = I_2$
Tiwari's method: $\bar{Q} = I_4$; $R = I_2$

To achieve the first and second design goals we add a forward-gain matrix H as shown in Fig. 1. The H can be solved from the block C_1 in eqn. 19d or

$$H = \omega_n^2 \begin{bmatrix} 14{\cdot}98 & 95150 \\ 85{\cdot}2 & 124000 \end{bmatrix}^{-1} = \begin{bmatrix} -198{\cdot}42349 & 152{\cdot}258 \\ 0{\cdot}136336 & -0{\cdot}023971 \end{bmatrix} \qquad (34)$$

Thus the design system is

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_1(s) \\ R_2(s) \end{bmatrix}$$

$$= \frac{90000}{s^2 + 450s + 90000} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_1(s) \\ R_2(s) \end{bmatrix} \qquad (35)$$

For this real nontrivial system the designed system is not only static decoupling but also complete noninteracting, and the final values of the unit-step responses are unity. The peak time is 0·014 s and the maximum percentage overshoot is 1%. The simulation curves for unit-step input are shown in Figs. 2 and 3. Comparing the design results of the proposed method with those of McMorran[14] and Tiwari *et al.*,[15] the present result gives less overshoot and less oscillatory responses.

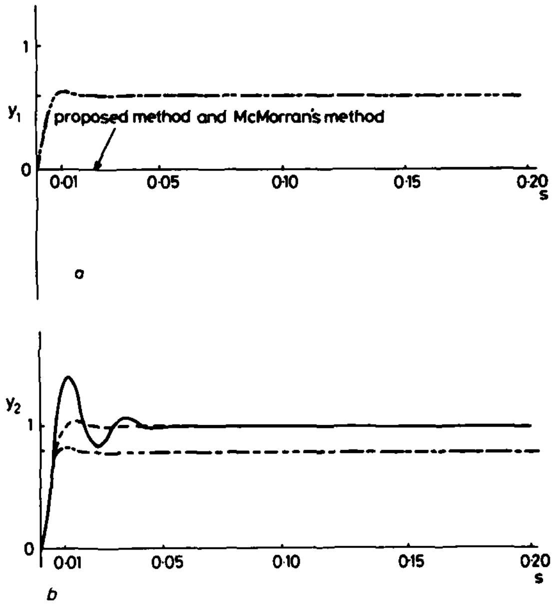


Fig. 3
Responses of various designed systems to a unit step in r_2

$r_1 = 0$
$r_2 = 1$
——— McMorran's method
- - - - proposed method: $\xi = 0{\cdot}75$; $\omega_n = 300$
— — — proposed method: $\bar{Q} = I_4$; $R = I_2$
Tiwari's method: $\bar{Q} = I_4$; $R = I_2$

5 Conclusion

A new method, based on a state equation in the phase-variable block co-ordinate, has been presented to determine the optimal block controllers for a class of multivariable systems. The reverse process of obtaining the optimal block controllers has been used to determine the weighting matrices of the quadratic performance index.

When a multivariable dynamic system is formulated in a matrix differential equation, the proposed method is more suitable for the determination of the optimal controllers than the conventional approach. Also, it is simpler to determine the weighting matrices than the conventional approaches. However, the proposed method is limited to a class of multivariable systems whose state equations can be formulated into matrix differential equations or the state equations in the block companion form.

6 Acknowledgments

The authors wish to express their gratitude for the valuable remarks and suggestions of the referees. This work was supported in part by US Army Missile Research and Development Command, DAAK 40-78-C-0017, DAAK 40-79-C-0061, and US Army Research Office DAAG 29-77-G-0143.

7 References

1 WELLS, D.A.: 'Lagrangian dynamics' (Schaum, New York, 1967), pp. 1–8
2 ANDERSON, B.D.O., and MOORE, J.B.: 'Linear optimal control', (Prentice-Hall, New Jersey, 1971), pp. 50–115
3 GIBSON, J.E., and REKASIUS, Z.V.: 'A set of standard specifications for linear automatic control systems', *Trans. Amer. Inst. Elect. Engrs.*, 1961, **80**, pp. 65–77
4 KIRK, D.E.: 'Optimal control theory' (Prentice-Hall, New Jersey, 1970), pp. 123–227
5 ELSGOLC, L.E.: 'Calculus of variations' (Addison-Wesley, Mass., 1962)
6 ANDERSON, B.D.O.: 'An algebraic solution to the spectral factorization problem', *IEEE Trans.*, 1967, **AC-12**, pp. 410–414
7 BLACK, K.F., and DENMAN, E.D.: 'Spectral factorization of proper matrix polynomials., *Int. J. Electron.*, 1977, **42**, pp. 569–579
8 KALMAN, R.E.: 'When is a linear control system optimal?' *J. Basic Eng.*, 1964, **86**, pp. 51–60
9 CHANG, S.S.L.: 'Synthesis of optimal control systems' (McGraw-Hill, New York, 1961)
10 TYLER, J.S., and TUTEUR, F.B.: 'The use of a quadratic performance index to design multivariable control systems', *IEEE Trans.*, 1966, **AC-11**, pp. 84–92
11 MOLINARI, B.P.: 'The stable regulator problem and its inverse', *ibid.*, 1973, **AC-18**, pp. 454–459
12 ANDERSON, K.W., and SHANNON, G.F.: 'Computational methods for solving the inverse-suboptimal-control problem', *Proc. IEE*, 1975, **122**, (3), pp. 321–324
13 MUELLER, G.S.: 'Linear model of 2-shaft turbojet and its properties', *ibid.*, 1971, **118**, pp. 813–815
14 McMORRAN, P.D.: 'Design of gas-turbine controller using inverse Nyquist method', *ibid.*, 1970, **117**, pp. 2050–2056
15 TIWARI, R.N., PURKAYASTHA, P., and TIWARI, S.N.: 'Synthesis of stable and optimal controllers for a 2-shaft gas turbine', *ibid.*, 1977, **124**, (12), pp. 1243–1248
16 OGATA, K.: 'Modern control engineering' (Prentice-Hall, New Jersey, 1970), pp. 216–258

8 Appendix

Block linear transformation

Consider a class of completely controllable, linear, time-invariant, multi-input, multi-output system

$$\dot{x}_0(t) = A_0 x_0(t) + B_0 u(t) \tag{36a}$$

$$y(t) = C_0 x_0(t) \tag{36b}$$

where $A_0 \in R^{n\times n}$, $B_0 \in R^{n\times m}$, $C_0 \in R^{l\times n}$, $x_0(t) \in R^{n\times 1}$, $y(t) \in R^{l\times 1}$, $u(t) \in R^{m\times 1}$. Assume that $l, m < n$ and $n/m = k$ (an integer) and define $r = n - m$. By a linear transformation

$$x_0(t) = T_1 z_1(t) \tag{37}$$

We wish to construct a state equation in the controllable block companion form

$$\dot{z}_1(t) = A_1 z_1(t) + B_1 u(t) \tag{38a}$$

$$y(t) = C_1 z_1(t) \tag{38b}$$

where

$$A_1 = T_1^{-1} A_0 T_1 = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] = \left[\begin{array}{ccccc|c} O_m & I_m & O_m & O_m & \cdots & O_m \\ O_m & O_m & I_m & O_m & \cdots & O_m \\ \cdot & \cdot & \cdot & \cdot & \cdots & \cdot \\ O_m & O_m & O_m & O_m & \cdots & I_m \\ \hline -D_1 & -D_2 & -D_3 & -D_4 & \cdots & -D_k \end{array}\right] \tag{38c}$$

$$B_1 = T_1^{-1} B_0 = \begin{bmatrix} O_{r\times m} \\ I_{m\times m} \end{bmatrix}, C_1 = C_0 T_1 = [N_1, N_2, \ldots, N_k], \tag{38d}$$

$A_{11} \in R^{r\times r}, A_{12} \in R^{r\times m}, A_{21} \in R^{m\times r}$, and $A_{22} \in R^{m\times m}$.
The constant matrices $D_i \in R^{m\times m}$ and $N_i \in R^{l\times m}$ are called block elements and the matrix $I_m = I_{m\times m} \in R^{m\times m}$ is an identity matrix. The matrices $O_m = O_{m\times m} \in R^{m\times m}$ and $O_{r\times m} \in R^{r\times m}$ are null matrices, respectively. The corresponding matrix transfer function of eqn. 38 can be directly formulated as

$$\begin{aligned} Y(s) &= [N_1 + N_2 s + \ldots + N_k s^{k-1}] [D_1 + D_2 s + \ldots \\ &\quad + D_k s^{k-1} + I_m s^k]^{-1} U(s) \\ &= N(s) D^{-1}(s) U(s) = T_r(s) U(s) \end{aligned} \tag{39}$$

where $U(s)$ and $Y(s)$ are Laplace transforms of $u(t)$ and $y(t)$, and $T_r(s)$ is a matrix transfer function.

The objective is to derive the linear-transformation matrix T_1 in eqn. 37. Because T_1 transforms a state equation in eqn. 36 to a block companion form in eqn. 38, T_1 is called as a block linear transformation. We further assume that the matrix B_0 in eqn. 36 can be partitioned into the form of $\left[\begin{array}{c} B_{11} \\ \hline B_{21} \end{array}\right]$ where $B_{21} \in R^{m\times m}$ is a non-singular matrix. This can be accomplished by rearranging the sequence of the elements in the state vector $x_0(t)$ in eqn. 36. By applying the first linear transformation

$$x_0(t) = K_1 x_1(t) \tag{40}$$

where

$$K_1 = \begin{bmatrix} I_{r\times r} & B_{11} \\ O_{m\times r} & B_{21} \end{bmatrix} \quad \text{and} \quad K_1^{-1} = \begin{bmatrix} I_{r\times r} & -B_{11}B_{21}^{-1} \\ O_{m\times r} & B_{21}^{-1} \end{bmatrix}$$

we have

$$\dot{x}_1(t) = \bar{A}_1 x_1(t) + \bar{B}_1 u(t) \tag{41a}$$

$$y(t) = \bar{C}_1 x_1(t) \tag{41b}$$

where

$$\bar{A}_1 = K_1^{-1} A_0 K_1 = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix}, \quad \bar{B}_1 = K_1^{-1} B_0 = \begin{bmatrix} O_{r\times m} \\ I_{m\times m} \end{bmatrix},$$

$\bar{C}_1 = C_0 K_1$, $\bar{A}_{11} \in R^{r\times r}$, $\bar{A}_{12} \in R^{r\times m}$, $\bar{A}_{21} \in {}^{m\times r}$, and $\bar{A}_{22} \in R^{m\times m}$.

To obtain the required state equation in eqn. 38, we perform the second linear transformation

$$x_1(t) = K_2 z_1(t) \tag{42a}$$

where

$$K_2 = \begin{bmatrix} Q_1^{-1} & O_{r\times m} \\ -Q_2 Q_1^{-1} & I_{m\times m} \end{bmatrix}, \quad K_2^{-1} = \begin{bmatrix} Q_1 & O_{r\times m} \\ Q_2 & I_{m\times m} \end{bmatrix} \tag{42b}$$

and

$$[Q_1^T \mid Q_2^T] = [q_1, \ldots, q_r \mid q_{r+1}, \ldots, q_n]. \tag{42c}$$

T designates the transpose of the matrix. The unknown matrices $Q_1^T \in R^{r\times r}$ (with r column vectors q_i) and $Q_2^T \in R^{r\times m}$ (with m column vectors q_j) can be evaluated as follows.

From eqn. 42*a*, 41*a* and 38*c* we have the matrix equation

$$K_2^{-1}\bar{A}_1 = A_0 K_2^{-1} \tag{43a}$$

or

$$\begin{bmatrix} Q_1 & O_{r\times m} \\ Q_2 & I_{m\times m} \end{bmatrix} \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} Q_1 & O_{r\times m} \\ Q_2 & I_{m\times m} \end{bmatrix} \tag{43b}$$

Expanding eqn. 43*b* yields

$$\begin{aligned} Q_1\bar{A}_{11} &= A_{11}Q_1 + A_{12}Q_2 \\ Q_1\bar{A}_{12} &= A_{12} \end{aligned} \tag{43c}$$

and

$$\begin{aligned} Q_2\bar{A}_{11} + \bar{A}_{21} &= A_{21}Q_1 + A_{22}Q_2 \\ Q_2\bar{A}_{12} + \bar{A}_{22} &= A_{22} \end{aligned} \tag{43d}$$

Performing a transpose operation on eqn. 43*c* and substituting eqns. 38*c* and 42*c* into it, we have the following recursive formulas:

$$\bar{A}_{11}^T q_i = q_{m+i} \quad \text{for} \quad i = 1, 2, \ldots, r \tag{44a}$$

$$\bar{A}_{12}^T q_i = O_{m\times 1} \quad \text{for} \quad i = 1, 2, \ldots, r-m \tag{44b}$$

and

$$\bar{A}_{12}^T q_{r-m+i} = e^i \qquad \text{for} \quad i = 1,2,\ldots,m \tag{44c}$$

where e^i is the $m \times 1$ unit column vector whose ith element is unity, and all other elements are zeros. Eqn. 44 can be further simplified as follows:

(i) If $k = 2$ then

$$q_i = (\bar{A}_{12}^T)^{-1} e^i \qquad \text{for} \quad i = 1,2,\ldots,m \tag{45a}$$

and

$$q_{m+i} = \bar{A}_{11}^T q_i \qquad \text{for} \quad i = 1,2,\ldots,m \tag{45b}$$

(ii) If $k > 2$, then

$$q_i = \begin{bmatrix} \bar{A}_{12}^T & \\ \bar{A}_{12}^T & (\bar{A}_{11}^T)^1 \\ \cdot & \\ \bar{A}_{12}^T & (\bar{A}_{11}^T)^{k-3} \\ \bar{A}_{12}^T & (\bar{A}_{11}^T)^{k-2} \end{bmatrix}^{-1} \begin{bmatrix} O_{m \times 1} \\ O_{m \times 1} \\ \cdot \\ O_{m \times 1} \\ e^i \end{bmatrix} \quad \text{for } i = 1,2,\ldots,m \tag{45c}$$

and

$$q_{jm+i} = \bar{A}_{11}^T q_{(j-1)m+i} \quad \textit{for} \quad i = 1,2,\ldots,m \text{ and } j = 1,2,\ldots,k-1. \tag{45d}$$

When the square matrices in eqns. 45*a* and 45*c* are not singular, the q_i in eqn. 42 can be obtained. Note that the determination of q_i in eqn. 45 only involves one inversion of a matrix. Thus the transformation matrix T_1 in eqn. 37. which links the co-ordinates $x_0(t)$ in eqn. 36 and the required co-ordinates $z_1(t)$ in eqn. 38, is

$$x_0(t) = K_1 K_2 z_1(t) = T_1 z_1(t) \tag{46}$$

It is believed that the block linear transformation T_1 is new.

An illustrative example

Consider the dynamic equation of an actual gas-turbine system[13] which is completely controllable and observable.

$$\begin{aligned} \dot{x}_0(t) &= A_0 x_0(t) + B_0 u(t) \\ y(t) &= C_0 x_0(t) \end{aligned} \tag{47}$$

where

$$A_0 = \begin{bmatrix} -1{\cdot}268 & -0{\cdot}04528 & 1{\cdot}498 & 951{\cdot}5 \\ 1{\cdot}002 & -1{\cdot}957 & 8{\cdot}52 & 1240 \\ 0 & 0 & -10 & 0 \\ 0 & 0 & 0 & -100 \end{bmatrix}$$

$$B_0 = \begin{bmatrix} B_{11} \\ \hline B_{21} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \hline 10 & 0 \\ 0 & 100 \end{bmatrix} \quad C_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$n = 4$, $l = m = 2$, $r = n - m = 2$, and $k = n/m = 2$. The block companion form in eqn. 38, the corresponding matrix transfer function, of this system are required.

Applying the linear transformation in eqn. 40 yields the state equation in eqn. 41

$$\begin{aligned} \dot{x}_1(t) &= \bar{A}_1 x_1(t) + \bar{B}_1 u(t) \\ y(t) &= \bar{C}_1 x_1(t) \end{aligned} \tag{48}$$

where

$$\bar{A}_1 = \left[\begin{array}{c|c} \bar{A}_{11} & \bar{A}_{12} \\ \hline \bar{A}_{21} & \bar{A}_{22} \end{array}\right] = \left[\begin{array}{cc|cc} -1{\cdot}268 & -0{\cdot}04528 & 14{\cdot}98 & 95150 \\ 1{\cdot}002 & -1{\cdot}957 & 85{\cdot}2 & 124000 \\ \hline 0 & 0 & -10 & 0 \\ 0 & 0 & 0 & -100 \end{array}\right]$$

$$\bar{B}_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \hline 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \bar{C}_1 = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{array}\right]$$

and

$$K_1 = \left[\begin{array}{c|c} I_{r \times r} & B_{11} \\ \hline O_{m \times r} & B_{21} \end{array}\right] = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 100 \end{array}\right]$$

Applying the recursive algorithm in eqn. 45*a*, we have

$$q_1 = (\bar{A}_{12}^T)^{-1} e^1 = \begin{bmatrix} 14{\cdot}98 & 85{\cdot}2 \\ 95150 & 124000 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -1{\cdot}98423 \times 10^{-2} \\ 1{\cdot}52258 \times 10^{-2} \end{bmatrix}$$

$$q_2 = (\bar{A}_{12}^T)^{-1} e^2 = \begin{bmatrix} 14{\cdot}98 & 85{\cdot}2 \\ 95150 & 124000 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1{\cdot}36336 \times 10^{-5} \\ -2{\cdot}39708 \times 10^{-6} \end{bmatrix}$$

and

$$q_3 = \bar{A}_{11}^T q_1 = \begin{bmatrix} -1{\cdot}268 & 1{\cdot}002 \\ -0{\cdot}04528 & -1{\cdot}957 \end{bmatrix} \begin{bmatrix} -1{\cdot}98423 \times 10^{-2} \\ 1{\cdot}52258 \times 10^{-2} \end{bmatrix} = \begin{bmatrix} 4{\cdot}04164 \times 10^{-2} \\ -2{\cdot}88984 \times 10^{-2} \end{bmatrix}$$

$$q_4 = \bar{A}_{11}^T q_2 = \begin{bmatrix} -1{\cdot}268 & 1{\cdot}002 \\ -0{\cdot}04528 & -1{\cdot}957 \end{bmatrix} \begin{bmatrix} 1{\cdot}36336 \times 10^{-5} \\ -2{\cdot}39708 \times 10^{-6} \end{bmatrix} = \begin{bmatrix} -1{\cdot}96893 \times 10^{-5} \\ 4{\cdot}073763 \times 10^{-6} \end{bmatrix} \tag{49}$$

The transformation matrix K_2 in eqn. 42*b* is

$$K_2^{-1} = \left[\begin{array}{c|c} Q_1 & O_{r \times m} \\ \hline Q_2 & I_{m \times m} \end{array}\right] = \left[\begin{array}{cc|cc} -1{\cdot}98423 \times 10^{-2} & 1{\cdot}52258 \times 10^{-5} & 0 & 0 \\ 1{\cdot}36336 \times 10^{-5} & -2{\cdot}39708 \times 10^{-6} & 0 & 0 \\ \hline 4{\cdot}04164 \times 10^{-2} & -2{\cdot}88984 \times 10^{-2} & 1 & 0 \\ -1{\cdot}96893 \times 10^{-5} & 4{\cdot}073763 \times 10^{-6} & 0 & 1 \end{array}\right]$$

The block linear transformation T_1 in eqn. 46 is

$$x_0(t) = K_1 K_2 z_1(t) = T_1 z_1(t) \tag{50}$$

where

$$T_1 = \begin{bmatrix} 14{\cdot}98 & 95150 & 0 & 0 \\ 85{\cdot}2 & 124000 & 0 & 0 \\ 18{\cdot}5671 & -2622{\cdot}1 & 10 & 0 \\ -5{\cdot}21389 \times 10^{-3} & 136{\cdot}829 & 0 & 100 \end{bmatrix}$$

The required block companion form in eqn. 38 is

$$\begin{aligned} \dot{z}_1(t) &= A_1 z_1(t) + B_1 u(t) \\ y(t) &= C_1 z_1(t) \end{aligned} \tag{51}$$

where

$$A_1 = \left[\begin{array}{cc|cc} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \hline -18{\cdot}5671 & 2622{\cdot}1 & -11{\cdot}8567 & 262{\cdot}21 \\ 5{\cdot}214 \times 10^{-3} & -136{\cdot}83 & 0 & -101{\cdot}368 \end{array}\right]$$

$$
B_1 = \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ \hline 1 & 0 \\ 0 & 1 \end{array}\right] \quad C_1 = \left[\begin{array}{cc|cc} 14{\cdot}98 & 95150 & 0 & 0 \\ 85{\cdot}2 & 124000 & 0 & 0 \end{array}\right]
$$

The corresponding matrix transfer function in eqn. 39 is

$$
Y(s) = [N_1 + N_2 s]\,[D_1 + D_2 s + I_2 s^2]^{-1}\,U(s) \tag{52}
$$

where

$$
N_1 = \begin{bmatrix} 14{\cdot}98 & 95150 \\ 85{\cdot}2 & 124000 \end{bmatrix} \quad N_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}
$$

$$
D_1 = \begin{bmatrix} 18{\cdot}5671 & -2622{\cdot}1 \\ -5{\cdot}214 \times 10^{-3} & 136{\cdot}83 \end{bmatrix},
$$

$$
D_2 = \begin{bmatrix} 11{\cdot}8567 & -262{\cdot}21 \\ 0 & 101{\cdot}368 \end{bmatrix}, \quad I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
$$

Determination of Equivalent Dominant Poles and Zeros Using Industrial Specifications

LEANG-SAN SHIEH, MEMBER, IEEE, YING-JYI PAUL WEI, MEMBER, IEEE, HSI-ZEN CHOW, AND ROBERT E. YATES

***Abstract*—A graphical method and an analytical method are presented to determine the equivalent dominant poles and zeros of a system using assigned industrial specifications. A second-order transfer function with two poles and one finite zero is used to investigate the relationships between industrial specifications and the two poles and one finite zero. Also, it is used to verify the rule of the thumb obtained from Axelby's empirical results. A frequency response data matching method is proposed for fitting a low-order transfer function using the assigned industrial specifications that are obtained from a given high-order transfer function. Thus the equivalent dominant poles and zeros of a high-order system can be determined from the identified low-order model.**

I. Introduction

In the filter and compensator designs it is necessary and useful to have a rapid method or a simple graphical method to determine the poles and zeros that dominate the characteristics of the transient response. These poles and zeros are called the dominant poles and zeros that can be used to estimate the dynamic behavior of the system response. In the literature, the definitions of the dominant poles and zeros are ambiguous. For example, the dominant poles are commonly defined as the poles which are located near the imaginary axis (the $j\omega$ axis) or the poles which have the smallest absolute value when no significant zeros appear. Sometimes a pole P_1 is defined as the dominant pole [1] if $|P_i| > 6|P_1|$ where P_i are other system poles. The roles of dominant zeros that are often neglected in the literature become significant if the precise dynamic characteristics of a system in the transient state are required. The zeros not only contribute to the initial conditions of the transient response but also increase the bandwidth in the frequency domain; therefore, the roles of the zeros are as important as those of the poles.

As the technologies are progressing, the accurate description of many physical systems results in a high-order transfer function that consists of many clustery poles and zeros in the s plane. The poles near the $j\omega$ axis may not be dominant poles because the dominant effects on the transient response behavior of the poles are cancelled by the nearby zeros, and the system response may be characterized by the collective efforts of a group of clustery poles and zeros. This implies that the poles and zeros which are not near the $j\omega$ axis may dominate the characteristics of the system response. Therefore, the equivalent dominant poles and zeros, rather than the dominant poles and zeros obtained from the geometric locations in the s plane, become significant in the analysis and synthesis of a high-order system. Furthermore, the design goals and the nature of a high-order system are often characterized by a set of control specifications [2] (called the industrial specifications) that are commonly classified as 1) the time-domain specifications, for example, the rise time and the overshoot, 2) the frequency-domain specifications, for example, the bandwidth and the phase margin, 3) the complex-domain specifications, for example, the damping ratio and the undamped natural angular frequency or the equivalent poles and zeros in the s plane. If the relationships among the time-domain, frequency-domain specifications, and the equivalent poles and zeros (the complex-domain specifications) can be simply determined from a simple equation or a working graph, then the selected poles and zeros in the design of filters and compensators become meaningful, and the design processes can be greatly simplified.

In this paper, a graphical method and an analytical method are proposed to determine the equivalent dominant poles and zeros using assigned industrial specifications. First, relationships among various industrial specifications will be studied. A second-order transfer function having two poles and one finite zero is used as a basis for the investigation. Several working graphs and mathematical expressions are developed for the determination of the two dominant poles and one dominant zero using the assigned industrial specifications. Then the equivalent dominant poles and zeros of a high-order system are determined by a new dominant frequency-response data matching method. The equivalent dominant poles and zeros thus obtained satisfy the exact assigned industrial specifications.

II. The Relationships Among Various Industrial Specifications

In control system design, the design goals are usually expressed in terms of a set of industrial specifications. The placement of poles and zeros based upon the assigned specifications needs certain experiences. If the relationships among various industrial specifications can be determined, then nonconflicting industrial specifications can be assigned as design goals, and the meaningful dominant poles and zeros can be selected for filter and compensator designs. Thus an effective design method may be developed.

Manuscript received July 13, 1978; revised January 25, 1979. This work was supported in part by U.S. Army Missile Command, Redstone Arsenal, AL, DAAK 00-79-C-0061, and U.S. Army Research Office DAAG29-77-G-0143.

L. S. Shieh, Y. J. Wei, and H. Z. Chow are with the Department of Electrical Engineering, University of Houston, Houston, TX 77004.

R. E. Yates is with the Guidance and Control Directorate, U.S. Army Missile Research and Development Command, Redstone Arsenal, AL 35809.

0018-9421/79/0800-0125$00.75 © 1979 IEEE

An empirical study on the relationships among various industrial specifications has been conducted by Axelby [3]. The empirical rules or the rule of the thumb, which link the specifications in both time and frequency domains, are listed as follows:

$$M_t \simeq M_p \simeq \frac{1}{\sin \phi_m} \tag{1a}$$

where M_t is the maximum value of unit-step response, M_p is the maximum value of the closed-loop frequency response, and ϕ_m is the phase margin;

$$M_e \simeq \frac{1}{\omega_c} \tag{1b}$$

where M_e is the maximum value of the error of the unit-ramp function and ω_c is the gain crossover frequency;

$$\omega_p \simeq \omega_c \tag{1c}$$

where ω_p is the peak value frequency or the frequency when M_p occurs;

$$\dot{M}_t \simeq \omega_c \tag{1d}$$

where $\dot{M}_t$ is the maximum value of the unit-impulse response;

$$t_p \simeq \frac{3}{\omega_c} \tag{1e}$$

where t_p is the peak value time or the time when M_t occurs;

$$t_v \simeq \frac{1.8}{\omega_c} \tag{1f}$$

where t_v is the time when the maximum error of the ramp function with respect to its input occurs;

$$t_c \simeq \frac{1}{\omega_c} \tag{1g}$$

where t_c is the time when $\dot{M}_t$ occurs.

Other rules of the thumb according to Truxal [4] are listed as follows:

$$t_r \omega_b \simeq 0.6\pi \text{ to } 0.9\pi \tag{1h}$$

where t_r is the rise time or the time required for the response to go from 10 to 90 percent of its final value and ω_b is the bandwidth in rad/s;

$$t_d \simeq \frac{1}{K_v} \tag{1i}$$

where t_d is the delay time or the time required to reach 50 percent of its final value and K_v is the velocity error constant.

Some other analytical results that represent the relationships between the time-domain specifications (but not the frequency-domain specifications) and the complex-domain specifications have been developed and can be found in standard textbooks [5], [6]. The most commonly used function for investigating the relationships is

$$\frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{2}$$

where $Y(s)$ and $R(s)$ are the output and input functions, respectively, and ξ is the damping ratio and ω_n is the undamped natural angular frequency. From (2) we observe that the zero of the system is located at infinity, and is not a significant zero. Since the time-domain specifications are often used to define the characteristics of the transient behavior, the roles of zeros become significant. Therefore, a better model than that of (2) should be used to study the relationships among the industrial specifications. The transfer function of a unit-feedback system that has two poles and one finite zero is used as a basis for the investigation. The proposed closed-loop transfer function is then,

$$\frac{Y(s)}{R(s)} = T(s) = \frac{\tau\omega_n s + \omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} = \frac{b_1 s + b_2}{s^2 + a_1 s + a_2} = \frac{B(s)}{A(s)}$$

$$= \frac{\tau\left(\dfrac{s}{\omega_n}\right) + 1}{\left(\dfrac{s}{\omega_n}\right)^2 + 2\xi\left(\dfrac{s}{\omega_n}\right) + 1} = \frac{\tau s^* + 1}{(s^*)^2 + 2\xi s^* + 1} \tag{3}$$

where s^* is a normalized complex variable, a_i and b_i are constants, and $A(s)$ and $B(s)$ are two polynomials. The normalized poles and the original poles are at

$$s_1^* = -\xi + j\sqrt{1 - \xi^2} \qquad s_1 = -\xi\omega_n + j\omega_n\sqrt{1 - \xi^2}$$

$$s_2^* = -\xi - j\sqrt{1 - \xi^2} \qquad s_2 = -\xi\omega_n - j\omega_n\sqrt{1 - \xi^2} \tag{4a}$$

and the normalized zero and the original zero are at

$$s^* = -\frac{1}{\tau} \qquad s = -\frac{\omega_n}{\tau}. \tag{4b}$$

The open-loop transfer function $G(s)$ of the system in (3) is

$$G(s) = \frac{\tau\omega_n s + \omega_n^2}{s[s + (2\xi\omega_n - \tau\omega_n)]} = \frac{K_v\left(1 + \dfrac{s}{b}\right)}{s\left(1 + \dfrac{s}{a}\right)} \tag{5}$$

where $K_v = \omega_n/(2\xi - \tau)$ is the velocity error constant if $\tau < 2\xi$

$$a = (2\xi - \tau)\omega_n \quad \text{and} \quad b = \omega_n/\tau.$$

Comparing (2) and (3) we observe that a finite zero has been inserted in (3). The zero contributes the initial condition at the transient state, and it reduces the velocity error at the steady state. Also it provides an additional bandwidth in the frequency domain, which increases the phase margin and improves the stability of a system.

The derivations of the relationships among the industrial specifications are shown as the following seven relationships.

1) The Relationships Among M_t, t_p, ξ, ω_n, and τ: The unit-step response of the system in (3) gives

$$Y(s) = \frac{\tau\omega_n s + \omega_n^2}{s(s^2 + 2\xi\omega_n s + \omega_n^2)}. \tag{6a}$$

The inverse Laplace transform of $Y(s)$ results in

$$Y(t) = 1 - e^{-\xi\omega_n t}\left[\cos \omega_n\sqrt{1 - \xi^2}\,t + \frac{\xi - \tau}{\sqrt{1 - \xi^2}} \sin \omega_n\sqrt{1 - \xi^2}\,t\right]. \tag{6b}$$

Differentiating $y(t)$ with respect to t and setting the result equal to zero yields

$$t_p = \left(\pi + \tan^{-1} \frac{\tau\sqrt{1-\xi^2}}{\tau\xi - 1}\right) \Big/ \left(\omega_n \sqrt{1-\xi^2}\right). \quad (6c)$$

Substituting (6c) into (6b) and simplifying it gives the maximum value of the unit-step response

$$M_t = 1 + e^{-\xi\omega_n t_p}(\tau^2 - 2\tau\xi + 1)^{1/2}. \quad (6d)$$

2) The Relationships Among M_p, ω_p, ξ, ω_n, and τ: Applying Higgins and Siegel's complex variable differentiation method [7], we can solve the peak value frequency ω_p from the following equation:

$$R_e\left\{j\left[\frac{1}{B(s)}\frac{dB(s)}{ds} - \frac{1}{A(s)}\frac{dA(s)}{ds}\right]\right\}_{s=j\omega_p} = 0. \quad (7a)$$

Thus we have

$$\left.\begin{aligned} \omega_p &= \omega_n\sqrt{1-2\xi^2} \\ M_p &= 1/(2\xi\sqrt{1-\xi^2}) \end{aligned}\right\}, \quad \text{if } \tau = 0 \quad (7b)$$

and

$$\left.\begin{aligned} \omega_p &= \frac{\omega_n}{\tau}\left[-1 + \sqrt{(\tau^2+1)^2 - 4\tau^2\xi^2}\right]^{1/2} \\ M_p &= \frac{\tau^2}{\sqrt{2}}\left[\sqrt{(\tau^2+1)^2 - 4\xi^2\tau^2} - (\tau^2+1) \right. \\ &\quad \left. + 2\xi^2\tau^2\right]^{-1/2}. \end{aligned}\right\}, \quad \text{if } \tau \neq 0. \quad (7c)$$

3) The Relationships Among ϕ_m, ω_c, ξ, ω_n, and τ: Using the definitions of ϕ_m and ω_c,

$$\phi_m = \underline{/G(s)}\big|_{s=j\omega_c} + 180° \quad (8a)$$

and

$$|G(s)|_{s=j\omega_c} = 1 \quad (8b)$$

we have

$$\phi_m = \tan^{-1}\left[\frac{\left(\frac{\omega_c}{\omega_n}\right)\tau + (2\xi - \tau)\left(\frac{\omega_n}{\omega_c}\right)}{1 - (2\xi - \tau)\tau}\right] \quad (8c)$$

and

$$\omega_c = \omega_n\left[2\xi\tau - 2\xi^2 + \sqrt{(2\xi^2 - 2\xi\tau)^2 + 1}\right]^{1/2}. \quad (8d)$$

4) The Relationships Among t_v, M_e, ξ, ω_n, and τ: The error signal $e(t)$, which is the difference between the ramp input $r(t)$ and the time response $y(t)$ of the same input to the system in (3), is

$$e(t) = \frac{2\xi - \tau}{\omega_n} - \frac{1}{A\omega_n} e^{-\xi\omega_n t}\left[B\cos\omega_n\sqrt{1-\xi^2}\,t - C\sin\omega_n\sqrt{1-\xi^2}\,t\right] \quad (9a)$$

where

$$A = (1-\xi^2),\ B = (2\xi - \tau)(1-\xi^2),$$
$$C = (1 - 2\xi^2 + \tau\xi)\sqrt{1-\xi^2}.$$

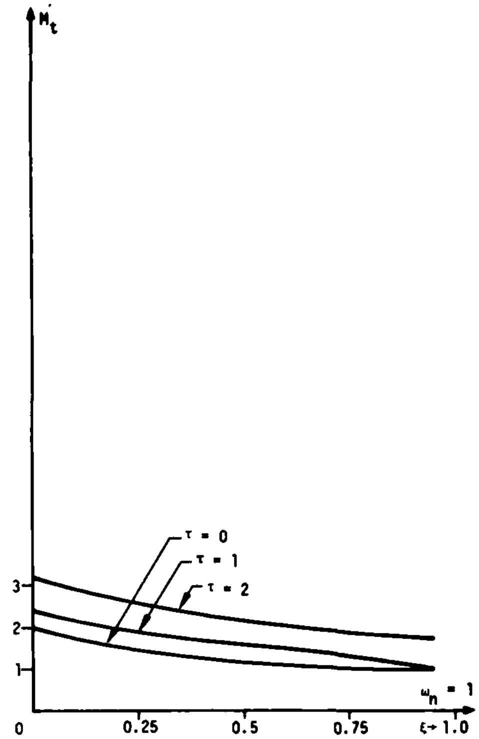


Fig. 1. Relationships among M_t, ξ, ω_n, and τ shown in (6d).

Differentiating $e(t)$ with respect to t and setting the result equal to zero we have

$$t_v = \frac{1}{\omega_n\sqrt{1-\xi^2}} \tan^{-1}\left[\frac{\sqrt{1-\xi^2}}{\tau - \xi}\right]. \quad (9b)$$

Substituting the t_v into (9a) and simplifying it we have

$$M_e = \left[2\xi - \tau + \sqrt{(1 + \tau^2 - 2\tau\xi)}e^{-\xi\omega_n t_v}\right]/\omega_n. \quad (9c)$$

5) The Relationships Among t_c, $\dot{M}_t$, ξ, ω_n, and τ: Differentiating the unit-impulse response $y(t)$ of the system in (3), $\dot{y}(t)$, and setting the result equal to zero, we have the time t_c at which the maximum value occurs, or

$$t_c = \frac{1}{\omega_n\sqrt{1-\xi^2}} \tan^{-1}\left[\frac{(1 - 2\xi\tau)\sqrt{1-\xi^2}}{\xi - 2\tau\xi^2 + \tau}\right]. \quad (10a)$$

Substituting t_c into $\dot{y}(t)$ yields the maximum value of the unit-impulse response $\dot{M}_t$, or

$$\dot{M}_t = \omega_n e^{-\xi\omega_n t_c}\sqrt{\tau^2 - 2\xi\tau + 1}. \quad (10b)$$

6) The Relationships Among K_v, ξ, ω_n, and τ: The velocity error constant K_v can be derived from the basic definition as

$$K_v = \lim_{s\to 0} s \cdot G(s) = \frac{\omega_n}{2\xi - \tau}, \quad \text{if } \tau < 2\xi. \quad (11)$$

7) The Relationships Among ω_b, ξ, ω_n, and τ: The definition of the bandwidth of a system is

$$|T(s)|_{s=j\omega_b} = \frac{1}{\sqrt{2}}. \quad (12)$$

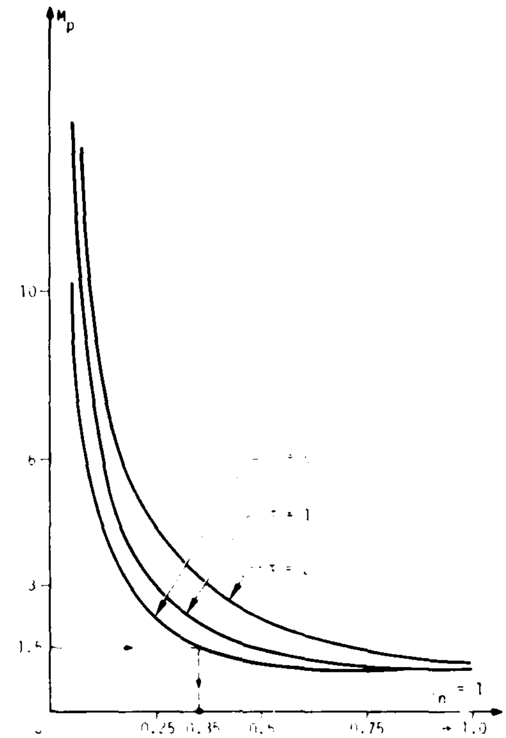


Fig. 2. Relationships among M_p, ξ, ω_n, and τ shown in (7).

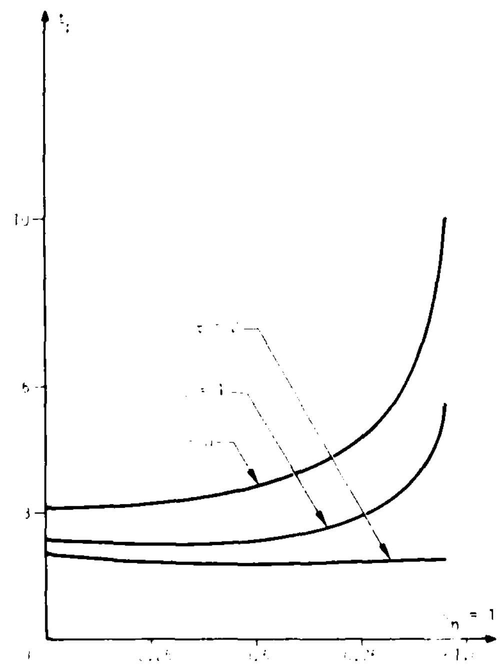


Fig. 3. Relationships among t_p, ξ, ω_n, and τ shown in (6c).

The analytical expression is

$$\omega_b = \omega_n \left[(1 + \tau^2 - 2\xi^2) + \sqrt{(1 + \tau^2 - 2\xi^2)^2 + 1}\right]^{1/2}. \qquad (13)$$

Most important time-domain and frequency-domain specifications have been analytically expressed in terms of ξ, ω_n, and

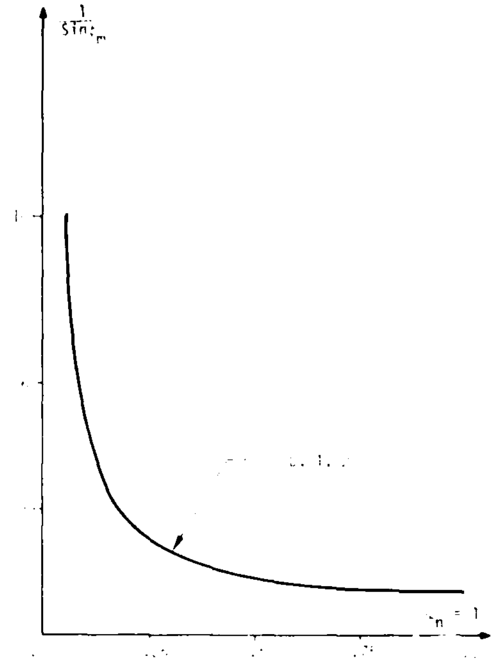


Fig. 4. Relationships among $1/\sin\phi_m$, ξ, ω_n, and τ shown in (8c).

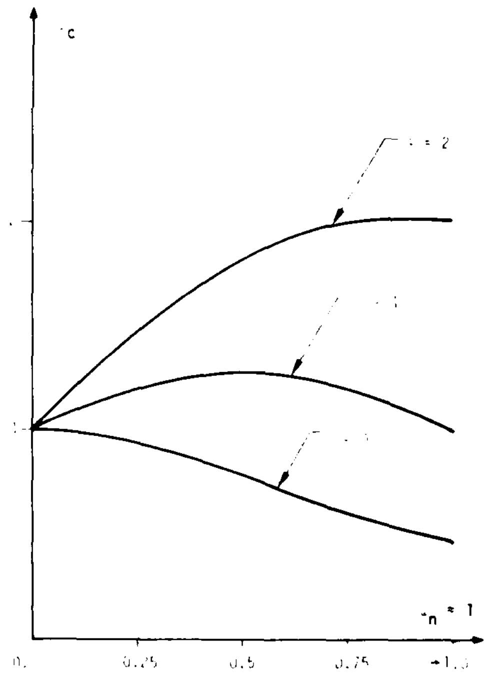


Fig. 5. Relationships among ω_c, ξ, ω_n, and τ shown in (8d).

τ which are the specifications in the complex domain. These expressions are normalized and graphically shown in Figs. 1–11. If an industrial specification is assigned, the corresponding ξ and τ or the equivalent poles and zero in (4) can be determined from the plotted curves. Also the curves in Figs. 12–15 can be used to verify the rules of the thumb proposed by Axelby [3]. It is observed that the accuracy of the rules de-

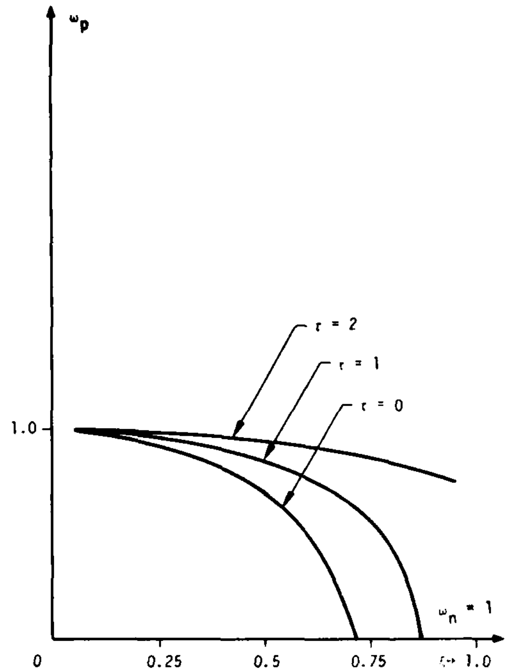


Fig. 6. Relationships among ω_p, ξ, ω_n, and τ shown in (7).

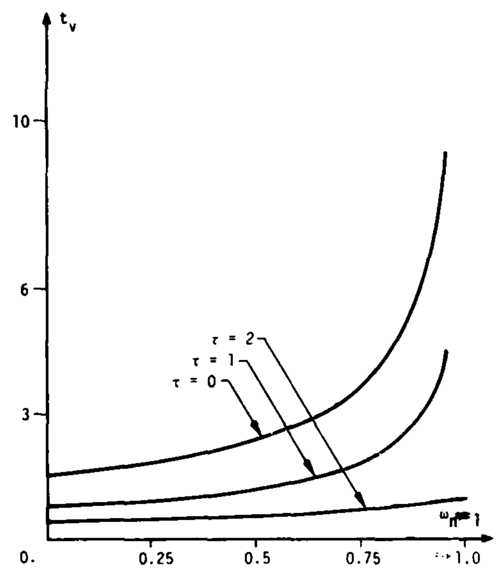


Fig. 7. Relationships among t_v, ξ, ω_n, and τ shown in (9b).

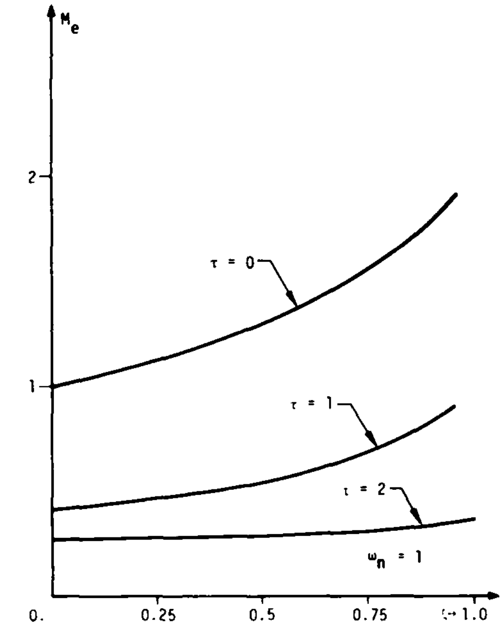


Fig. 8. Relationships among M_e, ξ, ω_n, and τ shown in (9c).

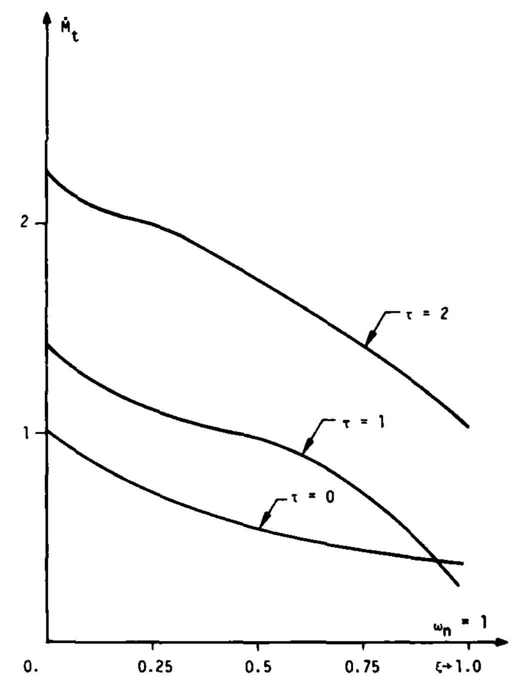


Fig. 9. Relationships among $\dot{M}_t$, ξ, ω_n, and τ shown in (10b).

pends upon the range of the damping ratio and the zero location. Furthermore, from the developed working graphs, a set of meaningful and nonconflicting specifications can be assigned for the design goals of a control system.

III. Determination of Equivalent Dominant Poles and Zeros from a High-Order Model

In the design of high performance control systems, quite often several specifications are assigned as design goals, and the corresponding dominant poles and zeros are required. This is a problem of a high-order transfer function fitting using industrial specifications. Shieh *et al.* [8], [9] have developed an original synthesis technique to fit a second-order transfer function based on three industrial specifications. The Newton-Raphson multidimensional method [10] was applied to solve the resulting nonlinear simultaneous equations that can be converted to a single variable quadratic equation. However, it is well known that the Newton-Raphson method will only con-

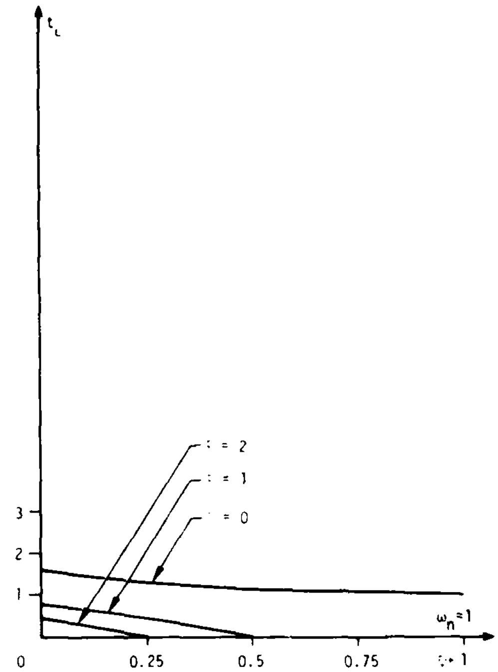


Fig. 10. Relationships among t_c, ξ, ω_n, and τ shown in (10a).

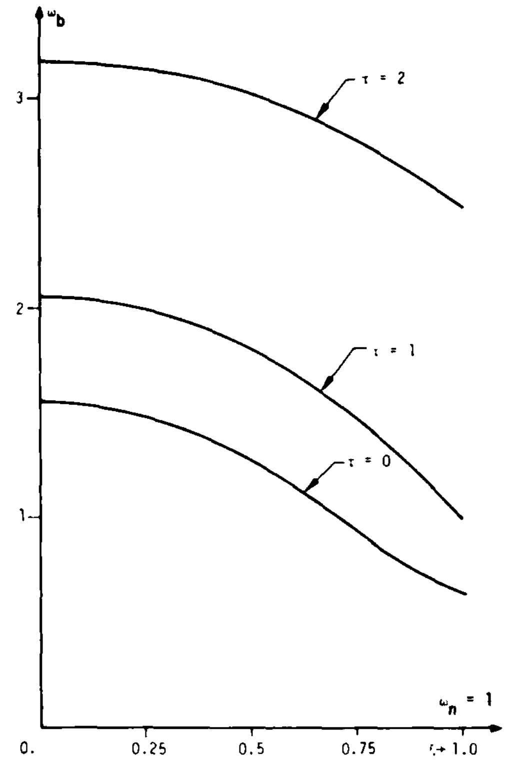


Fig. 11. Relationships among ω_b, ξ, ω_n, and τ shown in (13).

verge for a small range of starting values or the initial guesses. It is also known that high-order nonlinear equations have many solutions that depend heavily on the initial guess used. For general nonlinear equations that cannot be converted to a single variable equation, the Newton-Raphson numerical method may not converge to the desired solution using arbitrary initial guesses. In this paper, the original synthesis method [8], [9] is extended for modeling a high-order transfer function using many industrial specifications; and an analytical method is proposed for the estimation of the good starting values. Thus the desired dominant poles and zeros can be determined from the identified transfer function. The method can be well illustrated using the following example.

Suppose that the poles and zeros that represent the following given industrial specifications are required to be determined.

Type "1" system (14a)

ω_c the gain crossover frequency = 4.7 rad/s (14b)

ϕ_m the phase margin = 45.6° (14c)

M_p the maximum value of the closed-loop frequency response = 1.5 (14d)

ω_p the peak value frequency = 3.5 rad/s (14e)

ω_b the bandwidth of the closed-loop frequency response = 6.5 rad/s. (14f)

The assignments of the specifications in (14) closely follow the rules shown in (1). Therefore, the conflicted assignments can be avoided. The first two are the open-loop specifications, while the others are the closed-loop ones. Three equivalent poles and two equivalent zeros that represent the assigned specifications in (14) can be determined. The third-order model is

$$\frac{Y(s)}{R(s)} = T(s) = \frac{K(s+z_1)(s+z_2)}{(s^2+2\xi\omega_n s+\omega_n^2)(s+p)} = \frac{b_1 s^2 + b_2 s + b_3}{s^3 + a_1 s^2 + a_2 s + a_3} \tag{15a}$$

where K, p, ξ, ω_n, z_1, and z_2 or the corresponding a_i and b_i are unknown constants to be determined. Because the system is a type "1" system, the final value of the unit-step response of the system in (15a) is unity or

$$Y(t)|_{t\to\infty} = \lim_{s\to 0} s \cdot R(s)Y(s) = \lim_{s\to 0} s \cdot \left(\frac{1}{s}\right)\left(\frac{b_1 s^2 + b_2 s + b_3}{s^3 + a_1 s^2 + a_2 s + a_3}\right) = \frac{b_3}{a_3} = 1 \tag{15b}$$

or $$a_3 = b_3. \tag{15c}$$

As a result, (15a) can be simplified as

$$\frac{Y(s)}{R(s)} = T(s) = \frac{b_1 s^2 + b_2 s + a_3}{s^3 + a_1 s^2 + a_2 s + a_3}. \tag{15d}$$

The open-loop transfer function $G(s)$ is

$$G(s) = \frac{b_1 s^2 + b_2 s + a_3}{s[s^2 + (a_1 - b_1)s + (a_2 - b_2)]}. \tag{15e}$$

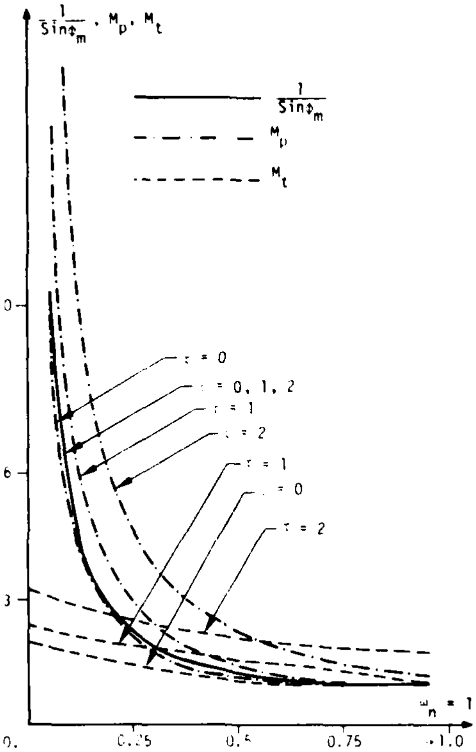


Fig. 12. Relationships among M_p, M_t, and $1/\sin\phi_m$ shown in (1).

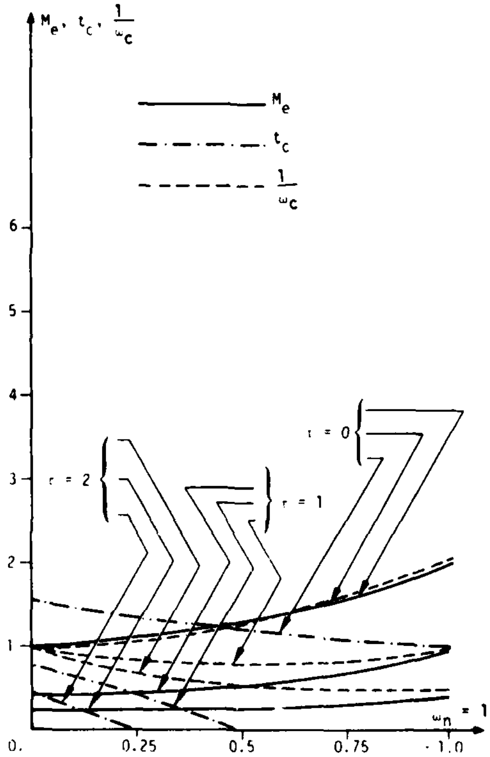


Fig. 13. Relationships among M_e, t_c, and $1/\omega_c$ shown in (1).

Following the definitions shown in (14), we can construct a set of nonlinear equations $f_i(a_1, a_2, a_3, b_1, b_2) = 0$ for $i = 1, 2, \cdots, 5$.

The definition of ω_c is

$$|G(j\omega_c)| = 1. \tag{16a}$$

The corresponding nonlinear equation is

$$f_1(a_1, a_2, a_3, b_1, b_2) = (a_1 - b_1)^2\omega_c^4 + [\omega_c^3 - (a_2 - b_2)\omega_c]^2 - (a_3 - b_1\omega_c^2)^2 - b_2^2\omega_c^2 = 0. \tag{16b}$$

The definition of ϕ_m can be expressed as

$$\phi_m = 180° + \underline{/G(j\omega_c)}. \tag{17a}$$

The nonlinear equation is

$$\begin{aligned} f_2(a_1, a_2, a_3, b_1, b_2) = {} & b_2\omega_c^2(a_1 - b_1) \\ & - (a_3 - b_1\omega_c^2)(\omega_c^2 - a_2 + b_2) \\ & - \tan\phi_m [(a_3 - b_1\omega_c^2)(a_1 - b_1)\omega_c \\ & + b_2\omega_c(\omega_c^2 - a_2 + b_2)] = 0. \end{aligned} \tag{17b}$$

The definition of ω_b is known as

$$|T(j\omega_b)| = \frac{1}{\sqrt{2}}. \tag{18a}$$

The corresponding nonlinear equation is

$$\begin{aligned} f_3(a_1, a_2, a_3, b_1, b_2) = {} & (a_3 - b_1\omega_b^2)^2 + b_2^2\omega_b^2 \\ & - \tfrac{1}{2}[(a_3 - a_1\omega_b^2)^2 + (\omega_b^3 - a_2\omega_b)^2] \\ = {} & 0. \end{aligned} \tag{18b}$$

The definition of ω_p gives

$$\left.\frac{d|T(j\omega)|}{d\omega}\right|_{\omega = \omega_p} = 0. \tag{19a}$$

Following Higgins and Siegel's complex variable differential technique [7], we have the following nonlinear equation:

$$\begin{aligned} f_4(a_1, a_2, a_3, b_1, b_2) = {} & [2a_1a_3\omega_p - 2a_1^2\omega_p^3 \\ & - (a_3 - 3\omega_p^2)(-\omega_p^3 + a_2\omega_p)] [(a_3 \\ & - b_1\omega_p^2)^2 + (b_2\omega_p)^2] + [-2a_3b_1\omega_p \\ & + 2b_1^2\omega_p^3 + b_2^2\omega_p] [(a_3 - a_1\omega_p^2)^2 \\ & + (-\omega_p^3 + a_2\omega_p)^2] = 0. \end{aligned} \tag{19b}$$

The definition of M_p is

$$|T(j\omega)|_{\omega = \omega_p} = M_p. \tag{20a}$$

The nonlinear equation is

$$\begin{aligned} f_5(a_1, a_2, a_3, b_1, b_2) = {} & (a_3 - b_1\omega_p^2)^2 + b_2^2\omega_p^2 \\ & - M_p^2[(a_3 - a_1\omega_p^2)^2 \\ & + (\omega_p^3 - a_2\omega_p)^2] = 0. \end{aligned} \tag{20b}$$

Equations (16)–(20) are a set of high-order nonlinear simultaneous equations which are very difficult to solve. The Newton-Raphson method, which is available in most digital computers

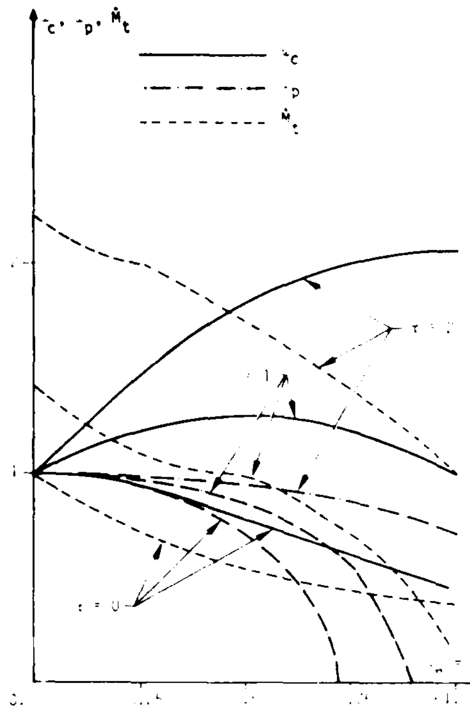


Fig. 14. Relationships among ω_c, ω_p, and $\hat{M}_t$ shown in (1).

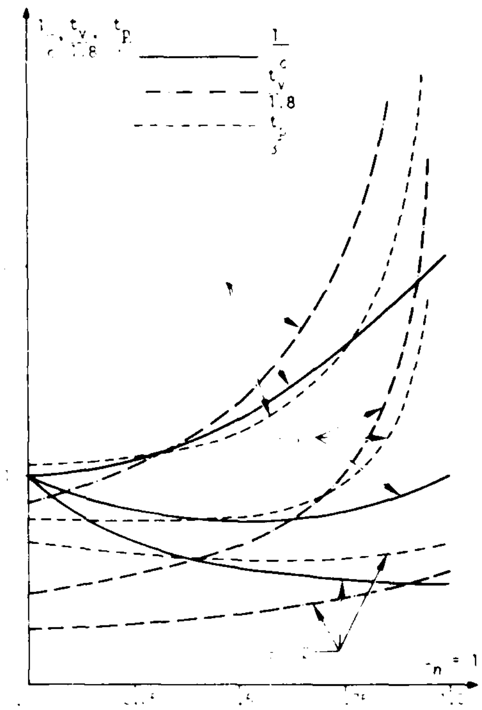


Fig. 15. Relationships among $1/\omega_c$, $t_r/1.8$, and $t_p/3$ shown in (1).

[11], is used to solve the nonlinear equations. To obtain the desired solution, and to improve the speed of convergence of the numerical method, we have to establish a set of good starting values. From the developed analytical expressions of various specifications or the working curves in this paper, we can determine the corresponding two poles and one zero using $M_p = 1.5$ and $\omega_p = 3.5$. From the rule of the thumb in (1) we observe that the M_p and ω_p have indirectly included the approximated respective ϕ_m and ω_c. The procedures are shown in the following steps.

Step 1: Determine the normalized dominant poles or the ξ in (4a) using the curve drawn in Fig. 2, having $\tau = 0$. From the curve ($\tau = 0$) we read the damping ratio $\xi = 0.35$. The normalized dominant poles and the dominant poles with $\omega_p = \omega_n = 3.5$ are

$$s_1^* = -0.35 + j0.9368 \qquad s_1 = -1.225 + j3.2786$$

$$s_2^* = -0.35 - j0.9368 \qquad s_2 = -1.225 - j3.2786. \tag{21a}$$

The second-order model is

$$T_2^*(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} = \frac{12.25}{s^2 + 2.45s + 12.25}. \tag{21b}$$

Step 2: Determine a dominant zero using the specification $\omega_b = 6.5$ in (14f). The modified second-order model becomes

$$T_2(s)^{**} = \frac{b_1 s + \omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} = \frac{b_1 s + 12.25}{s^2 + 2.45s + 12.25}. \tag{21c}$$

The b_1 can be easily determined by using the definition of ω_b in (18a):

$$b_1 = 3.1781. \tag{21d}$$

Thus a low-order dominant model is determined. However, a third-order model is required. An extra pole and a nearby zero are inserted into the second-order model in (21c) to obtain an approximate third-order model, or

$$T_3^*(s) = \frac{(b_1 s + \omega_n^2)(1.1s + 10\xi\omega_n)}{(s^2 + 2\xi\omega_n s + \omega_n^2)(s + 10\xi\omega_n)}$$

$$= \frac{3.49591s^2 + 52.406725s + 150.0625}{s^3 + 14.7s^2 + 42.2625s + 150.0625}. \tag{21e}$$

Using the coefficients in (21e) as initial guesses: $a_1^* = 14.7$, $a_2^* = 42.2625$, $a_3^* = 150.0625$, $b_1^* = 3.49591$, and $b_2^* = 52.406725$, and applying the Newton-Raphson method [11] to solve the nonlinear equations in (16) through (20) yields the desired solutions: $a_1 = 4.267162$, $a_2 = 20.58799$, $a_3 = 29.806197$, $b_1 = 3.188355$, and $b_2 = 15.561058$, at 10th iteration with the error tolerance of 10^{-6}. The desired transfer function is

$$T_3(s) = \frac{3.188355s^2 + 15.561058s + 29.806197}{s^3 + 4.267162s^2 + 20.58799s + 29.806197}. \tag{22}$$

The dominant poles and zeros, which represent the assigned industrial specifications, are determined from the poles P_i and zeros z_i in (22):

$$P_1 = -1.849412756$$

$$P_2 = -1.208824622 + j3.828226318$$

$$P_3 = -1.208824622 - j3.828226318 \tag{23a}$$

and

$$z_1 = -4.880591402 + j3.68424378$$

$$z_2 = -4.880591402 - j3.68424378. \tag{23b}$$

When the distribution of the poles and zeros of a high-order transfer function is known and the reduced-order transfer function that consists of equivalent dominant poles and zeros is required, it is a model reduction problem. Recently, various model reduction methods [12]-[15] have been proposed in the frequency domain. However, their reduced models [12]-[15], do not keep the assigned industrial specifications, which are obtained from the original system. The preservation of the exact frequency-domain specifications is essential in the design of filters and compensators using frequency-domain methods [5], [6], such as the Nyquist, Bode, and Nichols chart methods. This proposed method can overcome the shortcomings of the existing model reduction methods. The frequency-response data at ω_p, ω_b, ω_c, and ω_π (the phase crossover frequency of the open-loop system for the use of the gain margin [5], [6] are considered as the dominant frequency-response data. If some of these data are assigned to determine the corresponding reduced-order model, the equivalent dominant poles and zeros can be determined from the reduced-order model that consists of the exact industrial specifications assigned.

IV. Conclusion

A second-order transfer function with two poles and one finite zero has been used to derive the analytical and graphical expressions of various industrial specifications. For a few assigned industrial specifications, the corresponding two dominant poles and one dominant zero can be determined from the identified transfer function. The generalized second-order model has been used to verify the rule of the thumb proposed by Axelby. It has been observed that the accuracy of the rule of the thumb depends on the range of the damping ratio and the zero location. From the developed graphical expressions, a set of meaningful industrial specifications can be chosen and assigned as the design goals for the filter and compensator designs. A dominant frequency-response data matching method has been developed to construct a low-order transfer function using the assigned industrial specifications that are obtained from a given high-order system. Thus the equivalent dominant poles and zeros of a high-order system can be determined from the identified low-order transfer function that has the exact industrial specifications assigned.

More over, the proposed method in this paper has been successfully applied to redesign the compensators of a stabilized pitch control system of a real semiactive terminal homing missile [16]. The overall system characteristics of the redesigned missile [17] match those of the lower ordered model obtained from assigned industrial specifications.

References

[1] A. L. Greensite, *Elements of Modern Control Theory*. New York: Spartan Books, pp. 33-53, 1970.

[2] J. E. Gibson and Z. V. Rekasius, "A set of standard specifications for linear automatic control systems, "*AIEE Trans. Application and Industry*, pp. 65-77, May 1961.

[3] G. S. Axelby, "Practical methods of determining feedback control loop performance," in *Proc. 1st IFAC*, pp. 68-74, 1960.

[4] J. G. Truxal, *Control System Synthesis*. New York: McGraw-Hill, 1955, pp. 76-87.

[5] K. Ogata, *Modern Control Engineering*. Englewood Cliffs, NJ: Prentice-Hall, 1970, pp. 216-258.

[6] B. C. Kuo, *Automatic Control Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1975, pp. 459-500.

[7] T. J. Higgins and C. M. Siegel, "Determination of the maximum modulus, or of the specified gain, of a servomechanism by complex–differentiation," *IEEE Trans. Application and Industry*, pp. 467-468, 1953.

[8] L. S. Shieh, "An algebraic approach to system identification and compensator design," Ph.D. dissertation, Univ. of Houston, Houston, TX, 1970.

[9] C. F. Chen and L. S. Shieh, "An algebraic method for control system design," *Int. J. Contr.*, vol. 11. pp. 717-739, 1970.

[10] B. Carnahan, H. A. Luther, and J. O. Wilkes, *Applied Numerical Methods*. New York: Wiley, 1969, pp. 319-329.

[11] IBM S/370-360 Reference Manual IMSL, The International Mathematical and Statistical Library.

[12] C. F. Chen and L. S. Shieh, "A novel approach to linear model simplification," *Int. J. Contr.*, vol. 8, no. 6, pp. 561-570, 1968.

[13] L. S. Shieh and M. J. Goldman, "A mixed Cauer form for linear system reduction," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-4, pp. 584-588, Nov. 1974.

[14] M. F. Hutton and B. Friedland, "Routh approximations for reducing order of linear, time-invariant systems," *IEEE Trans. Automatic Contr.*, vol. AC-20, pp. 329-337, June 1975.

[15] Y. Shamash, "Linear system reduction using pade approximation to allow retention of dominant models," *Int. J. Contr.*, vol. 21, pp. 257-272, 1975.

[16] J. T. Bosley, "Digital realization of the T-6 missile analog autopilot," Final Rep., U.S. Army Missile Command, DAAK40-77-C-0048, TGT-001, May 1977.

[17] M. Datta-Barua, "Redesigning the stabilized pitch control system of a semiactive terminal homing missile." M. S. thesis. Univ. of Houston, Houston, TX, 1978.

INT. J. SYSTEMS SCI., 1979, VOL. 10, NO. 10, 1097–1114

A method for modelling transfer functions using dominant frequency-response data and its applications

L. S. SHIEH†, M. DATTA-BARUA†, and R. E. YATES‡

This paper presents a fundamental method for modelling transfer functions using the basic performance specifications and frequency-response dat . at the dominant frequencies. A set of non-linear equations is constructed from the definitions of the basic performance specifications, the dominant frequency-respon data and the unknown coefficients of a transfer function. A Newton–Raphson multidimensional method is applied to solve the non-linear equations. F ır methods are given to construct approximate representations of the desired transfer functions for the estimation of good starting values to ensure rapid convergence of the numerical method. The applications of the proposed method are: (1) developing a standard model and/or a transfer function of a filter or a compensator using the specified dominant frequency-response data; (2) identifying the transfer function of a system from available experimental frequency-response data; and (3) reducing high-order transfer functions to low-order models using dominant frequency-response data.

1. Introduction

The nature of the transient response of a system is often characterized by a set of performance specifications in the time domain such as the settling time and the rising time. In the frequency domain, another set of performance specifications (Gibson and Rekasius 1961) is used to represent the characteristics of the system performance. The bandwidth and the phase margin are typical examples of the frequency domain specifications. In designing compensators and filters, and in predicting the nature of time response of a system, practicing engineers are often interested in the dominant poles. These can be converted to a damping ratio and a natural angular frequency specified in the complex plane. These specifications are often called the complex-domain specifications. The engineer is also interested in various error constants (for example, the velocity-error constant), which represent the characteristics of system performance in both time and frequency domains (Truxal 1955). The frequency-response data at the frequencies of the frequency-domain specification are considered as the dominant frequency-response data in this paper because these data characterize the nature of the system responses. For example, the phase margin (ϕ_m) of a system at the gain-crossover frequency (ω_c) is often used as a measure of additional phase lag required to bring the system to the verge of instability. Also, if the phase angle of the open-loop system at the ω_c is near $-180°$, then the response of the closed-loop system will be oscillatory.

Received 5 July 1978.

† Department of Electrical Engineering, University of Houston, Houston, Texas 77004, U.S.A.

‡ Guidance and Control Directorate, U.S. Army Missile Research and Development Command, Redstone Arsenal, Alabama 35809, U.S.A.

0020-7721/79/1010 1097 \$02·00 © 1979 Taylor & Francis Ltd

In the design of a control system in the frequency domain, the specifications discussed above or the dominant frequency-response data are usually considered as design goals. Various frequency-domain or complex-domain approaches (Nyquist 1932, Evans 1953, Bode 1954, Thaler 1973) have been developed and widely applied in industry for compensator designs to achieve desired performance. The most popular design methods are those based on the Nyquist (1932) plot, the Bode (1954) design, and the root-locus method (Evans 1953, Thaler 1973). To improve the efficiency of the design methods, it is advantageous to have the design goals expressed as mathematical functions or transfer functions (defined as the standard models). Once standard models have been ascertained, the corresponding time-domain specifications and temporal responses can be determined from digital or analogue simulations of the standard models. Also, the frequency-response data of the desired compensator can be determined from Nyquist plots or Bode diagrams by comparing the frequency-response curves of the original and the desired response models. The required filters and compensators (Del Toro and Parker 1960, Thaler 1973) can then be easily determined.

Empirical rules or rules of the thumb that link the specifications in the time, frequency, and complex domains have been developed by Truxal (1955), Del Toro and Parker (1960), Axelby (1960), and Seshadri (1969) *et al.* From these results, it is observed that most time-domain specifications and complex-domain specifications can be approximately converted to frequency-domain specifications. Some of these frequency-domain specifications are phase margin (ϕ_m), maximum value of the closed-loop frequency response (M_p), gain-crossover frequency (ω_c), peak value frequency (ω_p), the bandwidth (ω_b), and velocity-error constant (K_v). Other important frequency-response data are : (1) the real part of the open-loop transfer function $G(j\omega)$ at the phase-crossover frequency (ω_π) which has been used to define the gain margin (G_m) ; (2) the real part and imaginary part of the closed-loop function ($T(s)$) and the open-loop function $G(s)$ at $s = j\omega \triangleq j\omega_0 = j0$. The data at $\omega = 0$ often indicate the final value and the type of the system. In a type 1 system, $I_m[G(j0)]$ has an infinite value, while Re $[G(j0)]$ has a finite value from which an asymptotic line (Del Toro and Parker 1960) can be drawn in a Nyquist plot ; (3) the corner frequencies in the Bode plot of $G(j\omega)$ in the regions of $\omega = \omega_{c1}$ where $20 \log |G(j\omega_{c1})| = +15$ dB, and $\omega = \omega_{c2}$ where $20 \log |G(j\omega_{c2})| = -15$ dB. Chen (1957) has shown empirically that the open-loop poles and zeros of a system can be approximated by retaining the Bode plot in the regions of the ± 15 dB boundaries. Some dominant frequency-response data are indicated in Fig. 1.

Through use of the above dominant frequency-response data, a basic method is proposed in this paper for modelling various transfer functions. First, a set of simultaneous non-linear algebraic equations, based on basic definitions of the dominant frequency-response data and the unknown coefficients of a desired transfer function, is constructed. Then the Newton–Raphson method (Carnahan *et al.* 1969, IBM 1977) is used to solve the non-linear equations. However, as is well known, the Newton–Raphson method will often only converge for a small range of starting values ; therefore, four methods are developed in this paper for estimating good starting values so that the numerical method (IBM 1977) will converge rapidly to the desired solution.

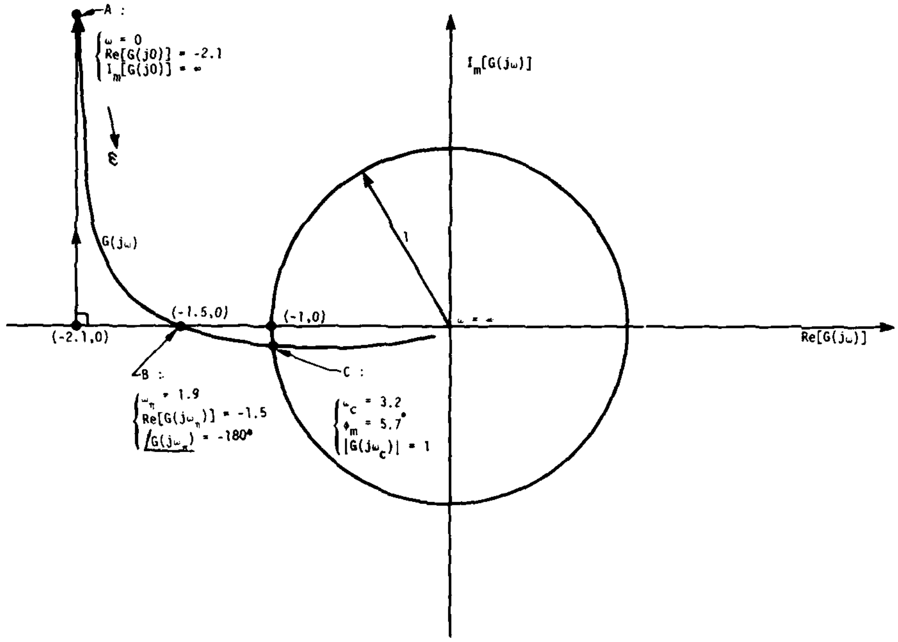


Figure 1. Nyquist plot of an *open-loop system* $G(s)$.

The applications of this method can be classified as follows.

(1) When the design goals are predescribed by the dominant frequency-response data, which may be obtained from the frequency-domain specifications (Gibson and Rekasius 1961) or equivalent ones (Truxal 1955, Del Toro and Parker 1960, Axelby 1960, Seshadzi *et al.* 1969), and a standard transfer function is desired, this is a *design problem.* Chen and Shieh (1970) and Wakeland (1976) have proposed analytical methods for the compensator fitting. However, their methods are limited to filters and compensators in which the unknown coefficients can be solved by a quadratic equation. The method of this paper overcomes this difficulty.

(2) The transfer function obtained in this paper is the function of the original system. When dominant frequency-response data can be obtained from experimental data of a practical system and the mathematical function of the system is desired, this is an *identification problem.*

(3) When the dominant frequency-response data are obtained from a given high-order transfer function and various low-order approximate models are required, this is the *model reduction problem.* The reduced models obtained in this paper have the same selected dominant frequency-response data as the original system. Thus, the design processes in the frequency domain can be greatly simplified.

2. Modelling non-linear equations

Given a transfer function $T(s)$ of a unity ratio feedback closed-loop system

$$T(s)=\frac{b_0+b_1s+b_2s^2+\ \dots\ +b_ms^m}{a_0+a_1s+a_2s^2+\ \dots\ +a_ns^n}=\frac{n(s)}{d(s)}=\frac{G(s)}{1+G(s)} \tag{1 a}$$

where $n(s)$ and $d(s)$ are the numerator and denominator polynomials, respectively, and a_i and b_i are constants. If the system is a type l system, the open-loop transfer function $G(s)$ is

$$G(s)=\frac{K(1+c_1s+c_2s^2+\ \dots\ +c_ps^p)}{s^l(1+d_1s+d_2s^2+\ \dots\ +d_qs^q)}=\frac{p(s)}{q(s)} \tag{1 b}$$

where $p(s)$ and $q(s)$ are the numerator and denominator polynomials. K, l, c_i, and d_i are constants. K is a velocity-error constant (K_v) if $l=1$.

The equations for dominant frequency-response data are :

(1) System type is determined from

$$G(j\omega_0)=\text{Re}\ [G(j\omega_0)]+jI_m[G(j\omega_0)] \quad \text{at } \omega_0=0 \tag{2 a}$$

or

$$\left.\begin{aligned} G(j0)&=\text{Re}\ [G(j0)] \\ T(j0)&=\frac{b_0}{a_0} \end{aligned}\right\} \text{ for a type 0 system} \tag{2 b}$$

$$\left.\begin{aligned} \text{Re}\ [G(j0)]&\approx K(c_1-d_1) \\ I_m[G(j0)]&\approx\infty \\ T(j0)&\approx\frac{b_0}{a_0}=1 \end{aligned}\right\} \text{ for a type 1 system} \tag{2 c}$$

(2) Phase margin gives

$$\phi_m=180^\circ+\angle G(j\omega_c) \tag{3 a}$$

where

$$|G(j\omega_c)|=1 \tag{3 b}$$

ω_c is the gain crossover-frequency.

(3) Gain margin yields

$$G_m=\left|\frac{1}{\text{Re}\ [G(j\omega_\pi)]}\right| \tag{4 a}$$

where

$$\angle G(j\omega_\pi)=-180^\circ \tag{4 b}$$

ω_π is the phase crossover frequency.

(4) $M_p=|T(j\omega_p)|$ = maximum value of the closed-loop frequency response (5 a)

where

$$\left.\frac{d|T(j\omega)|}{d\omega}\right|_{\omega=\omega_p}=0 \tag{5 b}$$

ω_p is the peak value frequency.

(5)
$$|T(j\omega_b)| = \frac{1}{\sqrt{2}} \tag{6}$$
where ω_b is the bandwidth.

(6)
$$|G(j\omega_{c1})| = 5{\cdot}6 \tag{7 a}$$
or
$$20 \log |G(j\omega)| = +15 \text{ dB} \quad \text{at } \omega = \omega_{c1} \tag{7 b}$$
and
$$|G(j\omega_{c2})| = 0{\cdot}18 \tag{7 c}$$
or
$$20 \log |G(j\omega)| = -15 \text{ dB} \quad \text{at } \omega = \omega_{c2} \tag{7 d}$$

A set of non-linear equations can be formulated from the basic definitions of the assigned dominant frequency-response data in (2)–(7). The procedures can be illustrated by using the following example. The dominant frequency-response data in (2 *c*), (3), and (4) are shown in Fig. 1, which are marked as A, B, and C and given as follows :

(1) $\text{Re}\,[G(j\omega_0)] = -2{\cdot}1$ and $I_m[G(j\omega_0)] = \infty$ at $\omega_0 = 0$ rad/s

or
$$T(j\omega_0) = 1 \quad \text{at } \omega_0 = 0 \text{ rad/s} \tag{8 a}$$

(2)
$$\text{Re}\,[G(j\omega_\pi)] = -1{\cdot}5 \quad \text{at } \omega_\pi = 1{\cdot}9 \text{ rad/s} \tag{8 b}$$

(3)
$$\angle G(j\omega_\pi) = -180° \quad \text{at } \omega_\pi = 1{\cdot}9 \text{ rad/s} \tag{8 c}$$

(4)
$$\phi_m = 180° + \angle G(j\omega_c) = 5{\cdot}7° \quad \text{at } \omega_c = 3{\cdot}2 \text{ rad/s} \tag{8 d}$$

(5)
$$|G(j\omega_c)| = 1 \quad \text{at } \omega_c = 3{\cdot}2 \text{ rad/s} \tag{8 e}$$

Five conditions are given in (8). Therefore, various transfer functions with five unknown coefficients can be constructed. Assume that the desired transfer function $T_d(s)$ is

$$T_d(s) = \frac{b_0 + b_1 s + b_2 s^2}{a_0 + a_1 s + a_2 s^2 + a_3 s^3} \tag{9 a}$$

From the conditions in (8 *a*), it may be observed that the system is a type 1 system. Therefore $b_0 = a_0$. Also, to simplify the equation we let $a_3 = 1$. Thus, we have

$$T_d(s) = \frac{a_0 + b_1 s + b_2 s^2}{a_0 + a_1 s + a_2 s^2 + s^3} \tag{9 b}$$

The corresponding open-loop transfer function $G_d(s)$ is

$$G_d(s) = \frac{K(1 + c_1 s + c_2 s^2)}{s(1 + d_1 s + d_2 s^2)} \tag{10}$$

where

$$K = \frac{a_0}{a_1 - b_1}, \quad c_1 = \frac{b_1}{a_0}, \quad c_2 = \frac{b_2}{a_0}, \quad d_1 = \frac{a_2 - b_2}{a_1 - b_1} \quad \text{and} \quad b_2 = \frac{1}{a_1 - b_1}$$

Following the basic definitions and the assigned data in (8) *yields a set of* non-linear equations :

(1) The assignment in (8 *a*), or Re $[G(j0)] = -2{\cdot}1$, gives

$$f_1(a_0, a_1, a_2, b_1, b_2) = a_1 b_1 - b_1^2 - a_0 a_2 + a_0 b_2 + 2{\cdot}1(a_1 - b_1)^2 = 0 \tag{11 a}$$

(2) The specification in eqn. (8 *b*), or Re $[G(j\omega_\pi)] = -1{\cdot}5$ at $\omega_\pi = 1{\cdot}9$, yields

$$\begin{aligned} f_2(a_0, a_1, a_2, b_1, b_2) &= (a_2 - b_2)(a_0 - 3{\cdot}61 b_2) - b_1(a_1 - b_1 - 3{\cdot}61) \\ &\quad - 1{\cdot}5[3{\cdot}61(a_2 - b_2)^2 + (a_1 - b_1 - 3{\cdot}61)^2] = 0 \end{aligned} \tag{11 b}$$

(3) The condition in (8 *c*), or $\angle G(j\omega_\pi) = -180°$ at $\omega_\pi = 1{\cdot}9$, gives

$$\begin{aligned} f_3(a_0, a_1, a_2, b_1, b_2) &= 3{\cdot}61 b_1(a_2 - b_2) \\ &\quad + (a_0 - 3{\cdot}61 b_2)(a_1 - b_1 - 3{\cdot}61) = 0 \end{aligned} \tag{11 c}$$

(4) The specification in (8 *d*), or $\phi_m = 5{\cdot}7°$ at $\omega_c = 3{\cdot}2$, yields

$$\begin{aligned} f_4(a_0, a_1, a_2, b_1, b_2) &= 10{\cdot}24 b_1(a_2 - b_2) + (a_0 - 10{\cdot}24 b_2)(a_1 - b_1 - 10{\cdot}24) \\ &\quad - 0{\cdot}319\,402\,24[(a_2 - b_2)(a_0 - 10{\cdot}24 b_2) \\ &\quad - b_1(a_1 - b_1 - 10{\cdot}24)] = 0 \end{aligned} \tag{11 d}$$

(5) The assignment in (8 *e*), or $|G(j\omega_c)| = 1$ at $\omega_c = 3{\cdot}2$, gives

$$\begin{aligned} f_5(a_0, a_1, a_2, b_1, b_2) &= (a_0 - 10{\cdot}24 b_2)^2 + 10{\cdot}24 b_1^2 \\ &\quad - 104{\cdot}8576(a_2 - b_2)^2 - 10{\cdot}24(a_1 - b_1 - 10{\cdot}24)^2 = 0 \end{aligned} \tag{11 e}$$

Equation (11) is a set of high-order simultaneous non-linear algebraic equations which are very difficult to solve. Considering the availability of the computer program package (IBM 1977) (called the Z systems) in many digital computers for the solution of non-linear equations, the Newton–Raphson multidimensional method is suggested for solving these equations. However, it is well known that the Newton–Raphson method will only converge for a small range of starting values or the initial guesses. A set of good initial guesses must be determined for rapid convergence of the numerical method. Four methods are proposed for these good initial guesses.

3. The initial guess

It is well known that high-order non-linear equations have many solutions. The solution and the speed of convergence of a numerical method depend heavily on the initial guesses or the starting values. In this paper, the Newton–Raphson method is suggested for solving the non-linear equations. The following methods, depending on the applications of interest, are proposed for good initial guesses.

3.1. *Initial guess by a synthesis method*

Suppose only the dominant frequency-response data in (8) are available and an approximate transfer function $T_d^*(s)$ of the desired $T_d(s)$ in (9 *b*) is required. The $T_d^*(s)$ is

$$T_d^*(s) = \frac{a_0^* + b_1^* s + b_2^* s^2}{a_0^* + a_1^* s + a_2^* s^2 + s^3} \tag{12}$$

where a_i^* and b_i^* are the starting values of the numerical method. The steps to obtain (12) are summarized as follows :

Step 1. Determine a second-order approximate transfer function $T_2^*(s)$ using $\phi_m = 5\cdot7°$ and $\omega_c = 3\cdot2$ rad/s in (8 *d*) and (8 *e*). This $T_2^*(s)$ is

$$T_2^*(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{13 a}$$

where ξ = the damping ratio and ω_n = the natural angular frequency. Two non-linear equations, which are constructed from the basic definitions of ω_c and ϕ_m, can be obtained. These non-linear equations can be converted into a single variable (ξ or ω_n) high-order equation from which the roots can be determined. Using this approach, we have $\xi = 0\cdot0498$ and $\omega_n = 3\cdot2079$. The poles that can be considered as the dominant poles of a system can be determined from the characteristic equation in (13 *a*). The dominant poles are

$$s_{1,2} = -\xi\omega_n \pm j\omega_n\sqrt{(1-\xi^2)} = -0\cdot1598 \pm j3\cdot2039 \tag{13 b}$$

Thus, (13 *a*) becomes

$$T_2^*(s) = \frac{10\cdot2909}{s^2 + 0\cdot3194s + 10\cdot2909} \tag{13 c}$$

Step 2. Construct a third-order approximate transfer function $T_3^*(s)$ by inserting in it a pole ($s = -p$) and modifying the term in the numerator of $T_2^*(s)$ so that the final value of the $T_3^*(s)$ equals to unity, or

$$T_3^*(s) = \frac{P\omega_n^2}{(s^2 + 2\xi\omega_n s + \omega_n^2)(s+P)} = \frac{10\cdot2909P}{(s^2 + 0\cdot3194s + 10\cdot2909)(s+P)} \tag{13 d}$$

The unknown constant P can be easily determined by using the condition in (8 *b*), or $\mathrm{Re}\,[G(j\omega_\pi)] = -1\cdot5$ where $\omega_\pi = 1\cdot9$. Thus, we have

$$P = 4\cdot5401 \tag{13 e}$$

Step 3. Establish another third-order approximate function $T_3^{**}(s)$ by inserting a zero in (13 *d*) with an unknown constant b_1^*.

$$T_3^{**}(s) = \frac{b_1^* s + P\omega_n^2}{(s^2 + 2\xi\omega_n s + \omega_n^2)(s+P)} = \frac{b_1^* s + 46\cdot7216}{(s^2 + 0\cdot3194s + 10\cdot2909)(s + 4\cdot5401)} \tag{13 f}$$

The b_1^* can be determined by using the condition in (2 *c*) and (8 *a*), or $\mathrm{Re}\,[G(j0)] = -2\cdot1$. The b_1^* is

$$b_1^* = 32\cdot4038 \tag{13 g}$$

Hence, we have

$$T_3^{**}(s) = \frac{46\cdot7216 + 32\cdot4038s}{46\cdot7216 + 11\cdot7410s + 4\cdot8595s^2 + s^3} \tag{13 h}$$

Equation (13 *h*) can be considered as an approximate function of (12) by assuming $b_2^* = 0$. The initial guesses in (12) are $a_0^* = 46\cdot7216$, $a_1^* = 11\cdot7410$, $a_2^* = 4\cdot8595$, $b_1^* = 32\cdot4038$, and $b_2^* = 0$. Using these constants as starting

values for the numerical method yields the desired coefficients in (9 *b*), or $a_0 = 6{\cdot}378\ 070$, $a_1 = 10{\cdot}462\ 220$, $a_2 = 1{\cdot}259\ 008$, $b_1 = 20{\cdot}556\ 61$, and $b_2 = 0{\cdot}243\ 466$. The desired transfer function is

$$T_3(s) = \frac{6{\cdot}378\ 070 + 20{\cdot}556\ 61s + 0{\cdot}243\ 466s^2}{6{\cdot}378\ 070 + 10{\cdot}462\ 220s + 1{\cdot}259\ 008s^2 + s^3} \tag{14}$$

The Newton–Raphson method (IBM 1977) converges at the 9th iteration with the error tolerance of 10^{-6}. Equation (14) has the exact frequency-response data specified in (8).

3.2. *Initial guess by complex-curve fitting and continued fraction methods*

The problem of finding unknown coefficients of a transfer function as a ratio of two frequency-dependent polynomials has been investigated by Levy (1959). His method minimizes the sum of squares of the errors at arbitrary experimental points. We present a simple method to determine the approximate coefficients of a transfer function using the real parts and imaginary parts of available limited frequency-response data. A low-order model is often determined because of data limitation. The low-order model is then expanded into a continued fraction of the Cauer second form to obtain a set of dominant quotients. Then some non-dominant quotients are inserted into the continued fraction to obtain an amplified-order model (Huang and Shieh 1976) which is the desired approximate transfer function for the use of the initial guess.

Consider the transfer function

$$T^*(s) = \frac{b_0 + b_1 s + b_2 s^2 + \dots + b_m s^m}{1 + a_1 s + a_2 s^2 + \dots + a_n s^n} \tag{15 a}$$

where a_i and b_i are unknown coefficients to be determined. Substituting $s = j\omega_k$ into (15 *a*) we have

$$\begin{aligned} T^*(j\omega_k) &= \frac{(b_0 - b_2\omega_k^2 + b_4\omega_k^4 - b_6\omega_k^6 + \dots) + j(b_1\omega_k - b_3\omega_k^3 + b_5\omega_k^5 - b_7\omega_k^7 + \dots)}{(1 - a_2\omega_k^2 + a_4\omega_k^4 - a_6\omega_k^6 + \dots) + j(a_1\omega_k - a_3\omega_k^3 + a_5\omega_k^5 - a_7\omega_k^7 + \dots)} \\ &= R(\omega_k) + jI(\omega_k) = R_k + jI_k \end{aligned} \tag{15 b}$$

when R_k and I_k are the given real and imaginary parts of the $T^*(s)$ at the available frequencies ω_k. Multiplying both sides of (15 *b*) by the common denominator and separating the real and imaginary parts, and also equating the respective real and imaginary parts, yields

$$\begin{aligned} b_0 - b_2\omega_k^2 + b_4\omega_k^4 - b_6\omega_k^6 + \dots + a_1 I_k\omega_k + a_2 R_k\omega_k^2 \\ - a_3 I_k\omega_k^3 - a_4 R_k\omega_k^4 + \dots = R_k \end{aligned} \tag{15 c}$$

and

$$\begin{aligned} b_1\omega_k - b_3\omega_k^3 + b_5\omega_k^5 - b_7\omega_k^7 + \dots - a_1 R_k\omega_k + a_2 I_k\omega_k^2 \\ + a_3 R_k\omega_k^3 - a_4 I_k\omega_k^4 + \dots = I_k \end{aligned} \tag{15 d}$$

In matrix form, (15 *c*) becomes

$$\begin{bmatrix} 1 & -\omega_1^2 & \omega_1^4 & -\omega_1^6 & . & I_1\omega_1 & R_1\omega_1^2 & -I_1\omega_1^3 & -R_1\omega_1^4 & . \\ 1 & -\omega_2^2 & \omega_2^4 & -\omega_2^6 & . & I_2\omega_2 & R_2\omega_2^2 & -I_2\omega_2^3 & -R_2\omega_2^4 & . \\ 1 & -\omega_3^2 & \omega_3^4 & -\omega_3^6 & . & I_3\omega_3 & R_3\omega_3^2 & -I_3\omega_3^3 & -R_3\omega_3^4 & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ 1 & -\omega_x^2 & \omega_x^4 & -\omega_x^6 & . & I_x\omega_x & R_x\omega_x^2 & -I_x\omega_x^3 & -R_x\omega_x^4 & . \end{bmatrix} \begin{bmatrix} b_0 \\ b_2 \\ b_4 \\ . \\ a_1 \\ a_2 \\ . \\ a_n \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ . \\ . \\ . \\ . \\ R_x \end{bmatrix} \qquad (15\,e)$$

where $x=n+m/2+1$ if m is even and $x=n+(m+1)/2$ if m is odd.

Substituting a_i obtained in (15 *e*) into (15 *d*), we have another matrix equation to solve for b_i, $i=1, 3, 5, \ldots$.

$$\begin{bmatrix} \omega_1 & -\omega_1^3 & \omega_1^5 & -\omega_1^7 & . & . & . \\ \omega_2 & -\omega_2^3 & \omega_2^5 & -\omega_2^7 & . & . & . \\ \omega_3 & -\omega_3^3 & \omega_3^5 & -\omega_3^7 & . & . & . \\ . & . & . & . & . & . & . \\ \omega_y & -\omega_y^3 & \omega_y^5 & -\omega_y^7 & . & . & . \end{bmatrix} \begin{bmatrix} b_1 \\ b_3 \\ b_5 \\ . \\ b_k \end{bmatrix}$$

$$= \begin{bmatrix} ((a_0I_1\omega_1^0+a_1R_1\omega_1^1)-(a_2I_1\omega_1^2+a_3R_1\omega_1^3)+\ \ldots) \\ ((a_0I_2\omega_2^0+a_1R_2\omega_2^1)-(a_2I_2\omega_2^2+a_3R_2\omega_2^3)+\ \ldots) \\ \\ ((a_0I_y\omega_y^0+a_1R_y\omega_y^1)-(a_2I_y\omega_y^2+a_3R_y\omega_y^3)+\ \ldots) \end{bmatrix} \qquad (15\,f)$$

where $\omega_k^0=1$, $a_0=1$; $k=m$ and $y=(m+1)/2$ if $m=$ odd; $k=m-1$ and $y=m/2$ if $m=$ even. In this example, the available data are

$$\left.\begin{aligned} &\omega_1=\omega_0=0, \quad R_1=T(j0)=1, \quad I_1=0 \\ &\omega_2=\omega_\pi=1{\cdot}9, \quad R_2=\mathrm{Re}\left[\frac{G(j\omega_\pi)}{1+G(j\omega_\pi)}\right]=2{\cdot}9684, \\ &\qquad\qquad I_2=I_{\mathrm{m}}\left[\frac{G(j\omega_\pi)}{1+G(j\omega_\pi)}\right]=-0{\cdot}0252 \\ &\omega_3=\omega_c=3{\cdot}2, \quad R_3=\mathrm{Re}\left[\frac{G(j\omega_c)}{1+G(j\omega_c)}\right]=0{\cdot}3351, \\ &\qquad\qquad I_3=I_{\mathrm{m}}\left[\frac{G(j\omega_c)}{1+G(j\omega_c)}\right]=-10{\cdot}4316 \end{aligned}\right\} \qquad (16)$$

Since only three values are available, the approximate function $T_2^*(s)$ is

$$T_2^*(s)=\frac{b_0+b_1s}{1+a_1s+a_2s^2} \qquad (17\,a)$$

Substituting the data at ω_1, and ω_2, and ω_3 in (16) into (15 *e*) yields $b_0=1$, $a_1=0{\cdot}0388$, and $a_2=0{\cdot}1839$. Then substituting a_i and the data at ω_3 into

(15 f) gives $b_1 = 2{\cdot}8907$. Because the desired approximate function in (12) is a third-order function, $T_2^*(s)$ should be amplified by using the continued fraction method (Huang and Shieh 1976) as follows.

$T_2^*(s)$ is first expanded into a continued fraction of the Cauer second form to obtain a set of dominant quotients : $h_1 = 1$, $h_2 = -0{\cdot}3507$, $h_3 = -0{\cdot}9651$, and $h_4 = 16{\cdot}0725$. Then the order of $T_2^*(s)$ is amplified to the third order by inserting non-dominant quotients $h_5 = 100$ and $h_6 = 0{\cdot}1$, or

$$T_2^*(s) = \frac{1 + 2{\cdot}8907s}{1 + 0{\cdot}0388s + 0{\cdot}1839s^2} = \cfrac{1}{h_1 + \cfrac{s}{h_2 + \cfrac{s}{h_3 + \cfrac{s}{h_4}}}} \simeq \cfrac{1}{h_1 + \cfrac{s}{h_2 + \cfrac{s}{h_3 + \cfrac{s}{h_4 + \cfrac{s}{h_5 + \cfrac{s}{h_6}}}}}}$$

$$= T_3^*(s) = \frac{54{\cdot}3885 + 162{\cdot}6914s + 15{\cdot}8219s^2}{54{\cdot}3885 + 7{\cdot}5839s + 10{\cdot}2146s^2 + s^3} \qquad (17\,b)$$

Huang and Shieh (1976) have shown that the amplified-order model is a good approximation of the original low-order model if the inserted positive quotients $h_i \gg 1$ and $h_{i+1} \ll 1$ where i is an odd number. Using the coefficients in (17 b) as initial guesses we have the desired coefficients in (14) at the 15th iteration (IBM 1977) with the error tolerance of 10^{-6}.

If much experimental frequency-response data, including the dominant data of a system, is available and the transfer function of the original system is required, this is an identification problem. In this case, a set of non-linear equations, based on the basic definitions of the dominant data, can be constructed and can be solved by the Newton–Raphson method. The initial guess can be determined by using the dominant data and others in (15). Since many data are available, a high-order approximate transfer function can be determined. Therefore, the use of the continued fraction method (Huang and Shieh 1976) is not necessary.

When a high-order transfer function of a system is given and various reduced-order transfer functions are required, this is a model reduction problem. In the frequency domain, numerous methods (Chen and Shieh 1969, Shieh and Goldman 1974, Hutton and Friedland 1975, Sharnash 1975, Lal and Van Valkenburg 1976) have been proposed for model reduction. The continued fraction methods (Chen and Shieh 1969, Shieh and Goldman 1974), the Routh approximation method (Hutton and Friedland 1975), the time-moment matching method (Shamash 1975), and the frequency-moment matching method (Lal and Van Valkenburg 1976) are the typical examples. These methods have been critically compared by Decoster and Cauwenberghe (1976). The new method presented in this paper can be used to obtain the reduced-order models which have the exact dominant frequency-response data as those of the original one. This method can be called a dominant frequency-response data matching method. The procedure is as follows.

Step 1. Plot the frequency-response curves to determine the data at the dominant frequencies ω_0, ω_π, ω_c, ω_{c1}, ω_{c2}, ω_p, and ω_b.

Step 2. Formulate a low-order model with unknown coefficients, and write a set of non-linear equations based on the basic definitions of the data at dominant frequencies.

Step 3. Determine a set of good starting values by using the synthesis method or the complex curve fitting method, and solve the non-linear equation by using the Newton–Raphson method. Thus, reduced-order models can be determined. Comparing the reduced-order models obtained from the proposed method with those of the existing methods (Chen and Shieh 1969, Shieh and Goldman 1974, Hutton and Friedman 1975, Shamash 1975, Lal and Van Valkenburg 1976), we observe that the model obtained in this paper is superior to existing methods in that the reduced model has the exact dominant frequency response as the original. As a result, an engineer can design a control system more efficiently in the frequency domain.

Since the original high-order transfer function is available, an existing method (Chen and Shieh 1969) can be applied and modified to obtain an approximate transfer function for the determination of the initial guess. Two additional methods for initial guess determination are as follows.

(3) Initial guess by a continued fraction method (Chen and Shieh 1969).

Consider the high-order transfer function in (1 a). The function can be expanded into a continued fraction and various reduced models obtained by discarding some of the quotients, or

$$T(s)=\frac{b_0+b_1s+\ \dots\ +b_ms^m}{a_0+a_1s+\ \dots\ +a_ns^n}=\frac{n(s)}{d(s)} \tag{18 a}$$

$$=\cfrac{1}{h_1+\cfrac{s}{h_2+\cfrac{s}{\ddots}}} \tag{18 b}$$

$$\simeq\frac{1}{h_1+\dfrac{s}{h_2}}=\frac{h_2}{h_1h_2+s} \tag{18 c}$$

$$\simeq\cfrac{1}{h_1+\cfrac{s}{h_2+\cfrac{s}{h_3+\cfrac{s}{h_4}}}}=\frac{h_2h_3h_4+(h_2+h_4)s}{h_1h_2h_3h_4+(h_1h_2+h_1h_4+h_3h_4)s+s^2} \tag{18 d}$$

$$\simeq\ \dots$$

Using the coefficients of the approximate model in (18) as the initial guess for the numerical method, we have the desired reduced model. However, the

approximate model in (18) may be unstable even if the original system is stable. The continued fraction method (Chen and Shieh 1969) can be modified by the following new method.

(4) Initial guess by a mixed method of the continued fraction approach and Gustafson's (1965) method.

Assume the reduced model of the original system in (18 *a*) is

$$T_p^*(s) = \frac{b_0^* + b_1^* s + \dots + b_{p-1}^* s^{p-1}}{a_0^* + a_1^* s + \dots + a_p^* s^p} = \frac{n^*(s)}{d^*(s)}, \quad a_p = 1 \tag{19 a}$$

A matrix equation (Chen and Shieh 1970) can be constructed from the dominant quotients h_i, $i = 1, 2, \dots, p$, obtained in (18 *b*) and the unknown coefficients a_i^* and b_i^* in (19 *a*) as

$$[b] = [H][a] \tag{19 b}$$

where

$$[a]^{\mathrm{T}} = [a_0^*, a_1^*, \dots, a_{p-1}^*] \tag{19 c}$$

$$[b]^{\mathrm{T}} = [b_0^*, b_1^*, \dots, b_{p-1}^*] \tag{19 d}$$

$$[H] = [H_2]^{-1}[H_1] \tag{19 e}$$

where T designates transpose,

$$[H_2] = \begin{bmatrix} h_1 & 0 & 0 & . & 0 & 0 \\ 1 & h_2 & 0 & . & 0 & 0 \\ 0 & 1 & h_3 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_p \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & h_1 & 0 & . & 0 & 0 \\ 0 & 1 & h_2 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_{p-1} \end{bmatrix} \cdots \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & 1 & 0 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ 0 & 0 & 0 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 0 & h_1 \end{bmatrix}$$

$$[H_1] = \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & h_2 & 0 & . & 0 & 0 \\ 0 & 1 & h_3 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_p \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & 1 & 0 & . & 0 & 0 \\ 0 & 0 & h_2 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 1 & h_{p-1} \end{bmatrix} \cdots \begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ 0 & 1 & 0 & . & 0 & 0 \\ 0 & 0 & 1 & . & 0 & 0 \\ 0 & 0 & 0 & . & 0 & 0 \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & 0 & h_2 \end{bmatrix}$$

The a_i^* in (19 *c*) can be determined from the coefficients of the polynomial that is obtained from the product of the dominant eigenvalues of the $d(s)$ in (18 *a*). When the dominant poles of $d(s)$ cannot be clearly identified or the poles of $d(s)$ are not available, the paper and pencil method suggested by Gustafson (1965) can be applied to construct the $d^*(s)$ or to determine a_i^* in (19 *c*). Then, substituting the a_i^* into (19 *b*) yields the required $n^*(s)$ or b_i^* in (19 *a*). The steps determine the $d^*(s)$ are shown as follows.

Step 1. Construct a Routh (1877) array using the coefficients a_i of $d(s)$ and the Routh algorithm. The a_i are expressed by double-subscripted notation $a_{i,j}$ for obtaining the general algorithm. The Routh array is

$$\left.\begin{array}{llll}
 & a_{11} \triangleq a_n & a_{12} \triangleq a_{n-2} & a_{13} \triangleq a_{n-4} \cdots a_0 \\
\gamma_1 = \dfrac{a_{11}}{a_{21}} & & & \\
 & a_{21} \triangleq a_{n-1} & a_{22} \triangleq a_{n-3} & a_{23} \triangleq a_{n-5} \cdots \\
\gamma_2 = \dfrac{a_{21}}{a_{31}} & & & \\
 & a_{31} \triangleq a_{12} - \gamma_1 a_{22} & a_{32} \triangleq a_{13} - \gamma_1 a_{23} & a_{33} \qquad \cdots \\
\gamma_3 = \dfrac{a_{31}}{a_{41}} & & & \\
 & a_{41} \triangleq a_{22} - \gamma_2 a_{32} & a_{42} \triangleq a_{23} - \gamma_2 a_{33} & \qquad \cdots \\
 & & & \\
 & a_{n-2,1} & a_{n-2,2} & \\
\gamma_{n-2} = \dfrac{a_{n-2,1}}{a_{n-1,1}} & & & \\
 & a_{n-1,1} & a_{n-1,2} = a_0 & \\
\gamma_{n-1} = \dfrac{a_{n-1,1}}{a_{n,1}} & & & \\
 & a_{n,1} & & \\
\gamma_n = \dfrac{a_{n,1}}{a_{n+1,1}} & & & \\
 & a_{n+1,1} = a_0 & &
\end{array}\right\} \quad (20\,a)$$

In general $a_{i,j} = a_{i-2,j+1} - \gamma_{i-2} a_{i-1,j+1}$; $i = 1, 2, \ldots, j = 3, 4, \ldots$

$$\gamma_i = a_{i,1}/a_{i+1,1} \qquad (20\,b)$$

Step 2. Construct various approximate low-order polynomials $d_i^*(s)$ from the last row and the next to last row, and so on in the Routh array.

For example, the ith order approximate equations are

$$d_1^*(s) = a_{n,1}s + a_{n+1,1} = a_{n,1}s + a_0 = 0 \quad \text{when } i = 1 \qquad (20\,c)$$

$$d_2^*(s) = a_{n-1,1}s^2 + a_{n,1}s + a_{n-1,2} = a_{n-1,1}s^2 + a_{n,1}s + a_0 = 0 \quad \text{when } i = 2 \qquad (20\,d)$$

and

$$\begin{aligned} d_3^*(s) &= a_{n-2,1}s^3 + a_{n-1,1}s^2 + a_{n-2,2}s + a_{n-1,2} \\ &= a_{n-2,1}s^3 + a_{n-1,1}s^2 + a_{n-2,2}s + a_0 = 0 \quad \text{when } i = 3 \end{aligned} \qquad (20\,e)$$

Since the original system is asymptotically stable, all γ_i are positive values. The approximate polynomials $d_i^*(s)$ are always the Hurwitz polynomials. Moreover, Gustafson (1965) has shown that relationships exist between the coefficients of $d_i^*(s)$ and the time-domain moments. The normalized polynomials can be determined by dividing each coefficient in $d_i^*(s)$ by the coefficient of the highest order term in s. The approximate transfer function $T_p^*(s)$

in (19 *a*) can be considered as a reduced-order model of the original high-order system. In this paper, we use it as the initial guess for the numerical method for determining the reduced order model that has the exact dominant frequency-response data as the original system.

4. An illustrative example

Consider the unit ratio feedback closed-loop transfer function of a stabilized real missile system (Bosley 1977)

$$T(s)=\frac{k'(b'_0+b'_1 s+ \dots +b'_5 s^5)}{a_0+a_1 s+ \dots +a_{11}s^{11}} \tag{21 a}$$

where

$$\begin{aligned}
a_0&=8{\cdot}802\ 158\ 509\times 10^{18}, & a_1&=2{\cdot}419\ 047\ 424\times 10^{19}\\
a_2&=2{\cdot}911\ 920\ 56\times 10^{18}, & a_3&=2{\cdot}420\ 405\ 431\times 10^{18}\\
a_4&=6{\cdot}667\ 397\ 031\times 10^{16}, & a_5&=9{\cdot}749\ 923\ 212\times 10^{14}\\
a_6&=9{\cdot}360\ 329\ 977\times 10^{12}, & a_7&=6{\cdot}231\ 675\ 318\times 10^{10}\\
a_8&=2{\cdot}976\ 950\ 696\times 10^{8}, & a_9&=9{\cdot}316\ 239\ 04\times 10^{5}\\
a_{10}&=1{\cdot}923\ 554\times 10^{3}, & a_{11}&=1
\end{aligned}$$

and

$$\begin{aligned}
k'&=1{\cdot}494\ 523\ 312\times 10^{11} & &\\
b'_0&=5{\cdot}889\ 609\ 375\times 10^{7}, & b'_1&=3{\cdot}084\ 598\ 703\times 10^{8}\\
b'_2&=1{\cdot}958\ 045\ 299\times 10^{7}, & b'_3&=3{\cdot}357\ 065\ 095\times 10^{5}\\
b'_4&=1{\cdot}715\ 193\ 3\times 10^{3}, & b'_5&=1
\end{aligned}$$

The second order and the third order reduced-order models which have some of the dominant frequency-response data of the original system are required. The open-loop transfer function $G(s)$ of the system is

$$G(s)=\frac{k(e_0+e_1 s+ \dots +e_5 s^5)}{s(g_0+g_1 s+ \dots +g_{10}s^{10})} \tag{21 b}$$

where

$$\begin{aligned}
g_0&=-2{\cdot}190\ 952\ 724\ 6\times 10^{19}, & g_1&=-1{\cdot}442\ 378\ 55\times 10^{16}\\
g_2&=2{\cdot}370\ 233\ 311\times 10^{18}, & g_3&=6{\cdot}641\ 763\ 067\times 10^{16}\\
g_4&=9{\cdot}748\ 428\ 689\times 10^{14}, & g_5&=9{\cdot}360\ 329\ 977\times 10^{12}\\
g_6&=6{\cdot}231\ 675\ 318\times 10^{10}, & g_7&=2{\cdot}976\ 950\ 696\times 10^{8}\\
g_8&=9{\cdot}316\ 239\ 04\times 10^{5}, & g_9&=1{\cdot}923\ 554\times 10^{3}\\
g_{10}&=1 & &
\end{aligned}$$

and

$$\begin{aligned}
k&=1{\cdot}494\ 523\ 312\times 10^{11} & &\\
e_0&=5{\cdot}889\ 609\ 375\times 10^{7}, & e_1&=3{\cdot}084\ 598\ 703\times 10^{8}\\
e_2&=1{\cdot}958\ 045\ 299\times 10^{7}, & e_3&=3{\cdot}357\ 065\ 095\times 10^{5}\\
e_4&=1{\cdot}715\ 193\ 3\times 10^{3}, & e_5&=1
\end{aligned}$$

Note that $G(s)$ is a non-minimum phase function ; its Nyquist plot is shown in Fig. 1. The dominant frequency-response data are chosen and given in (8). The set of non-linear equations are shown in (11). The initial guesses shown in (13 *b*) and (17 *b*) yields the required third-order reduced model in (14), or

$$T_3^*(s)=\frac{0{\cdot}243\ 466s^2+20{\cdot}556\ 61s+6{\cdot}378\ 07}{s^3+1{\cdot}259\ 008s^2+10{\cdot}462\ 22s+6{\cdot}378\ 07} \tag{22 a}$$

If the continued fraction method (Chen and Shieh 1969) in (18) is used, the approximate reduced model is

$$T_{3c}^*(s)=\frac{0{\cdot}6920s^2+19{\cdot}4692s+3{\cdot}7376}{s^3+0{\cdot}9488s^2+10{\cdot}1661s+3{\cdot}7376} \tag{22 b}$$

Using the coefficients in (22 *b*) as starting values for solving the non-linear equations in (11) yields the desired coefficients in (22 *a*) at the eighth iteration (IBM 1977) with the error tolerance of 10^{-} . If the mixed method in (19) and (20) is used, the normalized approximate denominator in (20 *e*) is

$$d_3^*(s)=s^3+0{\cdot}9524s^2+10{\cdot}1924s+3{\cdot}7455 \tag{22 c}$$

The $n_3^*(s)$ obtained from (19) is

$$n_3^*(s)=0{\cdot}7066s^2+19{\cdot}5155s+3{\cdot}7455 \tag{22 d}$$

The approximate transfer function by the mixed method is

$$T_{3m}^*(s)=\frac{0{\cdot}7066s^2+19{\cdot}5155s+3{\cdot}7455}{s^3+0{\cdot}9524s^2+10{\cdot}1924s+3{\cdot}7455} \tag{22 e}$$

If the coefficients in (22 *e*) are used as starting values, the Newton–Raphson method (IBM 1977) will converge to the desired solution in (22 *a*) at the eighth iteration with the error tolerance of 10^{-6}. The unit step response curves of various reduced models and the original system are compared in Fig. 2. All three reduced-order models give very satisfactory approximate time response curves. However, only the $T_3^*(s)$ in (22 *a*), which uses the method of dominant frequency-response data matching, has the exact dominant frequency-response data as the original system.

If $\omega_c=3{\cdot}2$ rad/s, $\phi_m=5{\cdot}7^\circ$ and $\mathrm{Re}\,[G(j0)]=-2{\cdot}4$ are chosen as the dominant data, the second-order reduced model obtained by the proposed method is

$$T_2^*(s)=\frac{3{\cdot}339\ 517s+9{\cdot}224\ 24}{s^2+0{\cdot}302\ 806s+9{\cdot}224\ 24} \tag{23 a}$$

The approximate reduced models by the continued fraction method and the mixed method are :

$$T_{2c}^*(s)=\frac{24{\cdot}7981s+4{\cdot}8122}{s^2+12{\cdot}8201s+4{\cdot}8122} \tag{23 b}$$

and

$$T_{2m}^*(s)=\frac{16{\cdot}3618s+3{\cdot}9328}{s^2+6{\cdot}5726s+3{\cdot}9328} \tag{23 c}$$

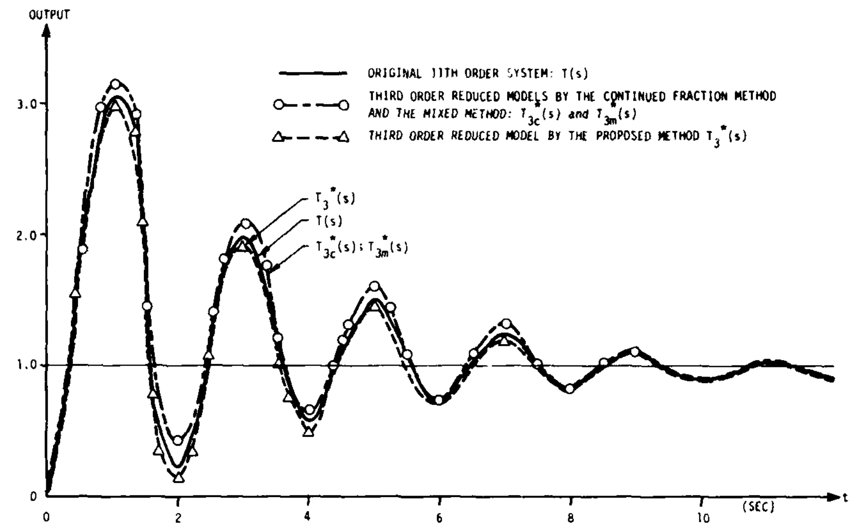


Figure 2. Time responses or original and third-order reduced models.

The unit-step time response curves of various reduced-order models $T_3{}^*(s)$, $T_2{}^*(s)$, $T_{2c}{}^*(s)$, and $T_{2m}{}^*(s)$ are compared in Fig. 3. It is observed that $T_2{}^*(s)$ gives better approximation in the transient response than $T_{2c}{}^*(s)$ and $T_{2m}{}^*(s)$.

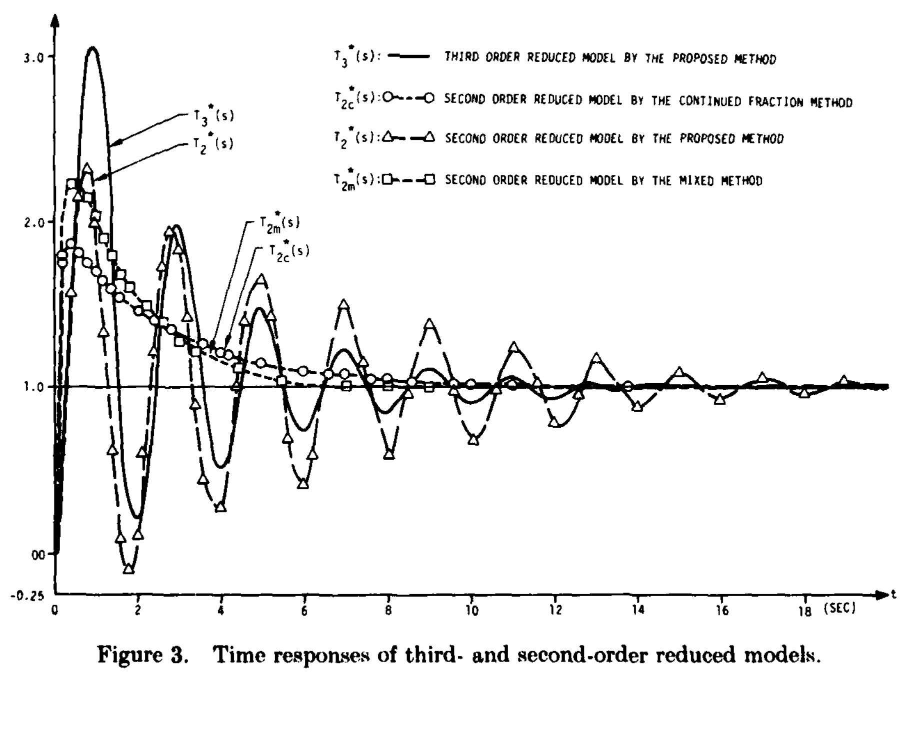


Figure 3. Time responses of third- and second-order reduced models.

5. Conclusion

A basic method has been developed for modelling transfer function using dominant frequency-response data. When the specifications of the design goals of a control system are assigned, the proposed method gives the standard transfer function. Thus, the design processes in the frequency domain can be significantly simplified. When the experimental frequency-response data of a system are available, the proposed method can be used to identify the transfer function of the original system. Also, if a high-order transfer function is given, various low-order models can be determined. The reduced models have the same dominant characteristics of the original system. Four methods have been proposed for estimating the good starting values for the solution of non-linear equations. The new dominant frequency-response data matching method, and the new mixed method that has the advantages of both continued fraction method of Chen and Shieh (1969) and the paper and pencil method of Gustafson (1965) have been developed for model reduction.

Acknowledgments

This work was supported in part by U.S. Army Missile Research and Development Command, DAAK 40-79-C-0061 and U.S. Army Research Office DAAG 29-77-G-0143.

References

Axelby, G. S., 1960, *Proc. 1st I.F.A.C.*, pp. 68–74.

Bode, H. W., 1954, *Network Analysis and Feedback Amplifier Design* (New York: Van Nostrand).

Bosley, J. T., 1977, U.S. Army Missile Command, DAAK40-77-C-0048, TGT-001.

Carnahan, B., Luther, H. A., and Wilkes, J. O., 1969, *Applied Numerical Methods* (New York: Wiley), pp. 319–329.

Chen, K., 1957, *A.I.E.E. Trans. Applic. Ind.*, **76,** 80.

Chen, C. F., and Shieh, L. S., 1968, *Int. J. Control*, **8,** 561; 1970, *Ibid.*, **11,** 717.

Decoster, M., and Van Cauwenberghe, A. R., 1976, *Journal* A, **17,** No. 2, 68; No. 3, 125.

Del Toro, V., and Parker, S., 1960, *Principles of Control Systems Engineering* (New York: McGraw-Hill), pp. 665–669, 278–302.

Evans, W. R., 1953, *Control System Dynamics* (New York: McGraw-Hill).

Gibson, J. E., and Rekasius, Z. V., 1961, *A.I.E.E. Trans. Applic. Ind.*, p. 65 (May, part II).

Gustafson, R. D., 1965, *Proc. JACC*, p. 301.

Huang, C. J., and Shieh, L. S., 1976, *Int. J. Systems Sci.*, **7,** 241.

Hutton, M. F., and Friedland, B., 1975, *I.E.E.E. Trans. autom. Control*, **20,** 329.

IBM, 1977, IBM S/370-360 Reference Manual IMSL (The International Mathematical and Statistical Library).

Lal, M., and Van Valkenburg, M. E., 1976, *9th Annual Asilomar Conference on Circuits, Systems, and Computers*, pp. 242–246.

Levy, E. C., 1959, *I.R.E. Trans. Autom. Control*, **4,** 37.

Nyquist, H., 1932, *Bell Syst. tech. J.*, **2,** 126.

Routh, E. J., 1877, *A Treatise on the Stability of a Given State of Motion* (London: Macmillan).

Seshadri, V., Rao, V. R., Eswaran, C., and Eappen, S., 1969, *Proc. Inst. elect. electron. Engrs*, **57,** 1321.

Shamash, Y., 1975, *Int. J. Control*, **21,** 257.

SHIEH, L. S., and GOLDMAN, M. J., 1974, *I.E.E.E. Trans. Syst. Man Cybernetics*, **4**, 584.
THALER, G. J., 1973, *Design of Feedback Systems* (Pa : Dowden, Hutchinson & Ross).
TRUXAL, J. G., 1955, *Control System Synthesis* (New York : McGraw-Hill), pp. 76–87.
WAKELAND, W. R., 1976, *I.E.E.E. Trans. autom. Control*, **21**, 771.